



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA PODNIKATELSKÁ
ÚSTAV INFORMATIKY

FACULTY OF BUSINESS AND MANAGEMENT
INSTITUTE OF INFORMATICS

TVORBA DATABÁZOVÉ APLIKACE A ŘEŠENÍ PRO BUSINESS INTELLIGENCE

CREATION OF DATABASE APPLICATION AND SOLUTIONS FOR BUSINESS INTELLIGENCE

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE
AUTHOR

Bc. MILAN MĚSTKA

VEDOUCÍ PRÁCE
SUPERVISOR

Ing. JIŘÍ KŘÍŽ, Ph.D.

BRNO 2012

ZADÁNÍ DIPLOMOVÉ PRÁCE

Městka Milan, Bc.

Informační management (6209T015)

Ředitel ústavu Vám v souladu se zákonem č.111/1998 o vysokých školách, Studijním a zkušebním řádem VUT v Brně a Směrnicí děkana pro realizaci bakalářských a magisterských studijních programů zadává diplomovou práci s názvem:

Tvorba databázové aplikace a řešení pro Business Intelligence

v anglickém jazyce:

Creation of Database Application and Solutions for Business Intelligence

Pokyny pro vypracování:

Úvod

Vymezení problému a cíle práce

Teoretická východiska práce

Analýza problému a současné situace

Vlastní návrhy řešení, přínos návrhů řešení

Závěr

Seznam použité literatury

Přílohy

Seznam odborné literatury:

HERNANDEZ M., VIESCAS J. Myslíme v jazyce SQL. 1. vydání. 2004. 380 s. ISBN 80-247-0899-X.

KOSEK J., XML pro každého. 1. vydání. 2000., 164s. ISBN 80-7169-860-1

NOVOTNÝ O., POUR J., SLÁNKSY D., Business Intelligence. Jak využít bohatství ve vašich datech. 1.

vydání. 2004. 256 s. ISBN 80-247-1094-3

PARR Olivia., Data mining - Praktickým průvodcem dolováním dat. 1. vydání. 2002. 330 s. ISBN 80-7226-577-6.

SHARP John., Microsoft visual C#. 1. vydání. 2010. 696 s. ISBN 978-80-251-3147-3.

Vedoucí diplomové práce: Ing. Jiří Kříž, Ph.D.

Termín odevzdání diplomové práce je stanoven časovým plánem akademického roku 2011/2012.

L.S.

Ing. Jiří Kříž, Ph.D.
Ředitel ústavu

doc. RNDr. Anna Putnová, Ph.D., MBA
Děkan fakulty

V Brně, dne 23.05.2012

Anotace

Tématem této diplomové práce je návrh databázové aplikace a řešení pro business intelligence v podniku. Návrh je realizován ve spolupráci s firmou ZZN Pelhřimov a.s. V úvodu práce jsou popsány teoretická východiska včetně základních pojmů a popisu vývojových prostředí (Sharp-develop, Visual Studio, MS Excel) ve kterém je projekt realizován. Dále je v kapitole 2 charakterizována firma ZZN a hlavní proces při kterém vznikají data pro analýzy. V hlavní části, je návrh řešení, způsob sběru dat a popis funkce jednotlivých částí softwarové podpory. V závěru se pomocí tohoto programu provedou analýzy, které jsou popsány v praktickém řešení a na jejich základě budou navrženy postupy pro zlepšení současné situace.

Annotation

Theme of this master's thesis is design of software support for business intelligence. Design is realized in cooperation with corporation ZZN Pelhřimov a.s. Introduction is focused on theoretical description of business intelligence and datamining and also on development environment in which is project designed. Corporation is characterised also in introduction. Main part contains data collecting and definition of individual modules. In conclusion of this thesis will be several types of analysis from collected data and then according to these analysis, we can draw measures to improve current state of corporation.

Klíčová slova

Bussines intelligence, Datamining, VBA, Visual Basic for applications , Excel, C sharp, Visual studio, Data, Sběr dat, Analýzy, podpora rozhodování, Databáze, Relační databáze, Datové sklady, ETL, SQL, Hvězda, OLTP, OLAP, Datová kostka, Multidimensionální, Microsoft.

Key words

Business intelligence, Datamining, VBA, Visual Basic for applications, Excel, C sharp, Visual studio, Data, Data collecting, Analysis, Decision support, Database, Relational diabase, Data warehouse, ETL, SQL, Star scheme, OLTP, OLAP, Datacube, Multidimensional, Microsoft.

Bibliografická citace

MĚSTKA, M. *Tvorba databázové aplikace a řešení pro business intelligence*. Brno: Vysoké učení technické v Brně, Fakulta podnikatelská, 2012. 81 s. Vedoucí diplomové práce Ing. Jiří Kříž, Ph.D.

Čestné prohlášení

Prohlašuji, že předložená diplomová práce je původní a zpracoval jsem ji samostatně.

Prohlašuji, že citace použitých zdrojů je úplná, že jsem v práci neporušil autorská práva
(v smyslu zákona č. 121/2000 Sb. O autorském právu a o právech souvisejících)

V Brně dne

.....
podpis

Poděkování

Rád bych poděkoval panu Ing. Jiřímu Křížovi, Ph.D. za kvalitní výuku a jeho pomoc při psaní této práce. Dále bych rád poděkoval společnosti ZZN Pelhřimov a.s., která mi poskytovala data a podklady ke zpracování.

OBSAH

ÚVOD	10
1. VYMEZENÍ PROBLÉMU A CÍLE PRÁCE	11
2. TEORETICKÁ VÝCHODISKA PRÁCE	12
2.1 Základní pojmy	12
2.1.1 Znalostní společnost	12
2.1.2 Globální ekonomika.....	13
2.1.3 IDE (Integrated development environment)	13
2.1.4 Relační databáze	14
2.2 Vývojové prostředí	15
2.2.1 Dot NET platforma (.NET).....	15
2.2.2 C# (C sharp).....	16
2.2.3 SharpDevelop.....	17
2.2.4 Visual Studio C#.....	17
2.2.5 SQL & MS SQL Server	18
2.3 Business intelligence.....	19
2.3.1 Základní vymezení.....	19
2.3.2 Business inteligence ve firmě	20
2.3.3 OLTP & OLAP	22
2.4 Nástroje Business intelligence	23
2.4.1 Datové sklady (Data Warehouse)	23
2.4.2 Datová tržiště (Data Mart)	31
2.4.3 Transformační nástroje (ETL)	31
2.4.4 Reporting	34
2.4.5 Multi-dimenzionální databáze	35
2.4.6 OLAP Datová kostka	36
2.4.7 Data-mining	38
3. ANALÝZA PROBLÉMU A SOUČASNÉ SITUACE	40
3.1 Charakteristika společnosti	40
3.2 Produkty firmy	41
3.3 Hlavní proces – EPC diagram.....	43
3.4 Současné využití BI ve společnosti	44
3.5 Návrh řešení	45
4. VLASTNÍ NÁVRHY ŘEŠENÍ	51
4.1 OLTP Aplikace	51
4.1.1 Datový adaptér (Navázání spojení s SQL serverem).....	51
4.1.2 Výběr správné databáze	52
4.1.3 Vytváření nových záznamů	55
4.1.4 Kontrola správnosti dat	56
4.1.5 Zobrazování uložených záznamů.....	57
4.1.6 Filtrace uložených záznamů.....	58

4.1.7 Editace uložených záznamů.....	58
4.2 Plnění OLTP Aplikace.....	59
4.2.1 Získání potřebných dat.....	59
4.2.2 Korekce dat	59
4.2.3 Načtení XLS souborů.....	62
4.2.4 Uložení dat z XLS souborů.....	63
4.3 ETL Fáze.....	64
4.3.1 Extrakce	64
4.3.2 Transformace	64
4.3.3 Nahrávání (Loading).....	65
4.3.4 Tabulky dimenzí	66
4.3.5 Tabulka faktů	66
4.4 Analýzy.....	67
4.4.1 Analýza geografického rozložení komodit	67
4.4.2 Analýza dodavatelů.....	68
4.4.3 Analýza firemních středisek	69
4.4.4 Analýza velikosti dodávek.....	70
4.4.5 Analýza kvality dodávaných komodit	71
4.4.6 Analýza vlhkosti v čase	72
4.5 Doporučení.....	73
4.5.1 Dodávka hnojiv, chemických přípravků a osiv	73
4.5.2 Uzavření obchodních dohod a smluv.....	74
4.5.3 Podnikové kurzy / obohacení podnikové znalostní báze	74
4.5.4 Nabídka služeb (Přeprava).....	74
4.5.5 Příprava pracovišť, technologie a pomocné síly.....	75
5. ZÁVĚR	76
6. SEZNAM ZDROJŮ	77
7. SEZNAM PŘÍLOH.....	79

ÚVOD

Téma diplomové práce „Návrh databázové aplikace a řešení pro business intelligence“ jsem si vybral proto, že při realizaci využiji znalostí z několika různých předmětů. Především to jsou tyto předměty: Datové a funkční modelování, Pokročilé metody programování, Datové sklady, Business intelligence a Kancelářské aplikace. Jazyk pro realizaci byl zvolen C# a platforma „dot NET“, pro databázi je to pak platforma MS SQL. Tyto prostředí jsou kompatibilní s operačními systémy používanými ve společnosti a jsou tak vhodným řešením. V následujících částech této práce vysvětlím všechny potřebné pojmy k pochopení dané problematiky a dále také moduly samotného programu.

1. VYMEZENÍ PROBLÉMU A CÍLE PRÁCE

Cílem této diplomové práce bude navrhnout a následně realizovat softwarový nástroj, pomocí něhož bude možné ukládat data do datového skladu a následně provést multi-dimenzionální analýzu. Nástroj by měl být schopen přejmout relačně orientovaná data z transakčních operací a zpracovat je. Na základě provedené analýzy pak vytvořit návrhy na zlepšení. Pro řešení této práce využiji metody a postupy, které jsem se naučil během svého studia. Zejména to budou metody algoritmizace a moderní programovací techniky.

2. TEORETICKÁ VÝCHODISKA PRÁCE

2.1 Základní pojmy

Než se začneme plně věnovat problematice „Business intelligence“ a „Data-miningu“, vysvětlíme si nejprve pár základních pojmů, které nám umožní detailněji pochopit tuto problematiku a poskytnou nám dobré základy pro další studium.

2.1.1 Znalostní společnost

Protože v dalších kapitolách budeme vysvětlovat, proč se začala používat metoda Business intelligence, musíme představit termín „Znalostní společnost“, neboť jedním z hlavních důvodů je právě ona.

Jak nám již název napovídá, budou zde hrát klíčovou roli znalosti. A jsou to právě znalosti, potažmo jejich kvalita, co nám zásadně ovlivňují život. Můžeme tedy prohlásit, že znalostní společnost je taková společnost, ve které jsou znalosti jedním z nejdůležitějších forem kapitálu. Ve stejném duchu můžeme tuto situaci přenést do ekonomického prostředí, tím pádem se tedy budeme bavit o znalostní ekonomice. Znalost se stává strategickým zdrojem pro ekonomický rozvoj a konkurenceschopnost organizací, podniků či celé společnosti. Ne však každá společnost může z dobrých znalostí získávat konkurenční výhodu. Existují společnosti, které jsou na kvalitu znalostí citlivější. Vezměme například firmu, která se zabývá vývojem softwarových aplikací, tato firma je téměř úplně závislá na znalostech svých zaměstnanců. Na druhé straně firma, která se zabývá například sestavováním mechanických částí čerpadel, není téměř vůbec závislá na znalostech dělníků, nýbrž na jejich zručnosti.

Současný trend se přiklání situaci, že každá firma má možnost z dobrých znalostí získat výhodu a to díky globální ekonomice. Tento pojem si vysvětlíme v následující kapitole.¹

¹ VERCELLIS, Business Intelligence: Data mining and optimization for decision making, s.12

2.1.2 Globální ekonomika

Jednotlivé ekonomiky vnímáme jako shrnutí hospodaření určitých subjektů, například jednotlivých států. Globální ekonomiku pak můžeme chápat jako integrovanou soustavu dílčích ekonomik. Jen těžko si lze představit každodenní život bez produktu a surovin z celého světa. Tedy obchodování a propojenost jednotlivých ekonomik je v dnešní době samozřejmostí. Tím se dostáváme k důležitosti již zmiňované znalostní ekonomiky. Pokud bychom se vrátili do minulosti, můžeme pozorovat, že dříve, například vlastnictví určité suroviny znamenalo absolutní výhodu oproti ostatním. S rozvojem globální ekonomiky se situace poněkud mění. Jednotlivé státy a ekonomiky mezi sebou směňují suroviny, které vlastní nebo služby ve kterých jsou lepší než ostatní. Tím se zajišťuje diverzifikace komodit a služeb a zároveň se tak snižují náklady. Vlastnictví určité suroviny tedy stále ještě poskytuje určitou výhodu, ale neznamena zaručený úspěch v daném odvětví. Tím pádem jsme došli k jasnému závěru, že jsou to právě znalosti a jejich správné užití co nás může dovést k získání konkurenční výhody.²

2.1.3 IDE (Integrated development environment)

S pojmem „IDE“ – Integrated development environment se setkáme v programování nebo při tvorbě webových stránek. V překladu tato zkratka znamená integrované vývojové prostředí. Tedy je to integrovaný systém různých nástrojů k vývoji softwaru nebo webových prezentací. K vytvoření softwarového produktu nepotřebujeme žádný drahý produkt, vystačíme si pouze s kompilátorem pro konkrétní jazyk. Při vytváření webových stránek dokonce nepotřebujeme ani kompilátor, vystačíme si i s nejobyčejnějším textovým editorem. Nicméně výrazné urychlení a usnadnění vývoje nabízejí právě zmíněné integrované prostředí. Základní komponenty IDE jsou:

Editor zdrojového kódu

Jedná se v podstatě o textový editor. Navíc má funkce jako zvýrazňovač syntaxe, Intelli-sense, což je návrh možných příkazů již během zadání několika prvních písmen nebo automatická korektura syntaxe. Všechny tyto funkce zpřijemňují a urychlují psaní zdrojového kódu.

² VERCELLIS, Business Intelligence: Data mining and optimization for decision making, s.12 - 17

Kompilátor

Další samozřejmou součástí je kompilátor, který překládá zdrojový kód do kódu strojového, se kterým již pracují jednotlivé části počítače. Většina integrovaných prostředí obsahuje hned několik kompilátorů pro různé programovací jazyky.

Debugger:

Tento nástroj slouží k odstraňování chyb ve zdrojovém kódu. Máme zde však na mysli chyby logické, ne syntaktické. Např. když nám vychází špatné číslo u matematické operace. Hledání takových chyb může být velice náročné, neboť sám program je nedokáže odhalit.

Editor uživatelského rozhraní

S příchodem objektově orientovaného programování se programy spouštějí v grafickém prostředí (oknech). Toto je nástroj pro jejich návrh.³

2.1.4 Relační databáze

Jelikož technologie datových skladů a nástroje business intelligence vycházejí z relačních databází, stručně se seznámíme i s nimi. Jako každá jiná databáze i relační databáze slouží k uchovávání dat. Z časového hlediska patří technologie relačních databází k těm mladším a první zmínky o relačních databázích se objevují v roce 1970.

Jako nejjednodušší databázi si můžeme představit obyčejný papírový seznam, ve kterém uchováváme informace o nějaké entitě např. o zákazníkovi. Celý seznam si tak můžeme představit jako tabulku, tuto terminologii volíme schválně, protože ji přímo aplikujeme v relačních databázích. V této tabulce budeme mít jednotlivé informace o entitě, v našem případě o zákazníkovi. Mohou to tedy být údaje o bydlišti, identifikačním čísle, jménu apod. Tyto údaje jsou tedy atributy dané entity a můžeme si je představit jako sloupce dané tabulky. Tímto způsobem můžeme uchovávat informace prakticky o čemkoliv. V reálném světě však spolu objekty a entity korespondují, např. zákazník si

³ [http://msdn.microsoft.com/en-us/library/ms173064\(v=vs.90\).aspx](http://msdn.microsoft.com/en-us/library/ms173064(v=vs.90).aspx) , Introduction to IDE

objednává určité produkty, objednávku vyřizují určití zaměstnanci apod. V praxi tedy vznikl požadavek tyto vztahy zachycovat a právě to relační databáze umožňují.

Na téma relačních databází by bylo možné psát celé knihy a do detailu popisovat všechny jejich vlastnosti, to však není našim cílem, a proto nám prozatím budou stačit tyto základní informace.⁴

2.2 Vývojové prostředí

K realizaci cíle této práce budou využívány nejrůznější vývojové nástroje a technologie. V této kapitole se seznámíme se všemi platformami a nástroji, které budeme využívat.

2.2.1 Dot NET platforma (.NET)

.NET platforma je vyvíjena společností Microsoft, jedná se o soubor technologií pro vývoj a spouštění aplikací v prostředí MS Windows. Tuto architekturu si můžeme představit jako další systém, který je spuštěn v operačním systému Windows. Platforma je podporována pro desktopové aplikace, webové aplikace a nyní také nově pro operační systém Windows Mobile, který se instaluje do mobilních zařízení a tabletů.⁵

Výhody

- Podpora několika programovacích jazyků (C#, Visual Basic, Python, F#, atd.)
- Objektově orientovaná platforma
- BCL (Base class library) – rozsáhlá knihovna funkcí
- Bezpečnost
- Management operační paměti

Nevýhody

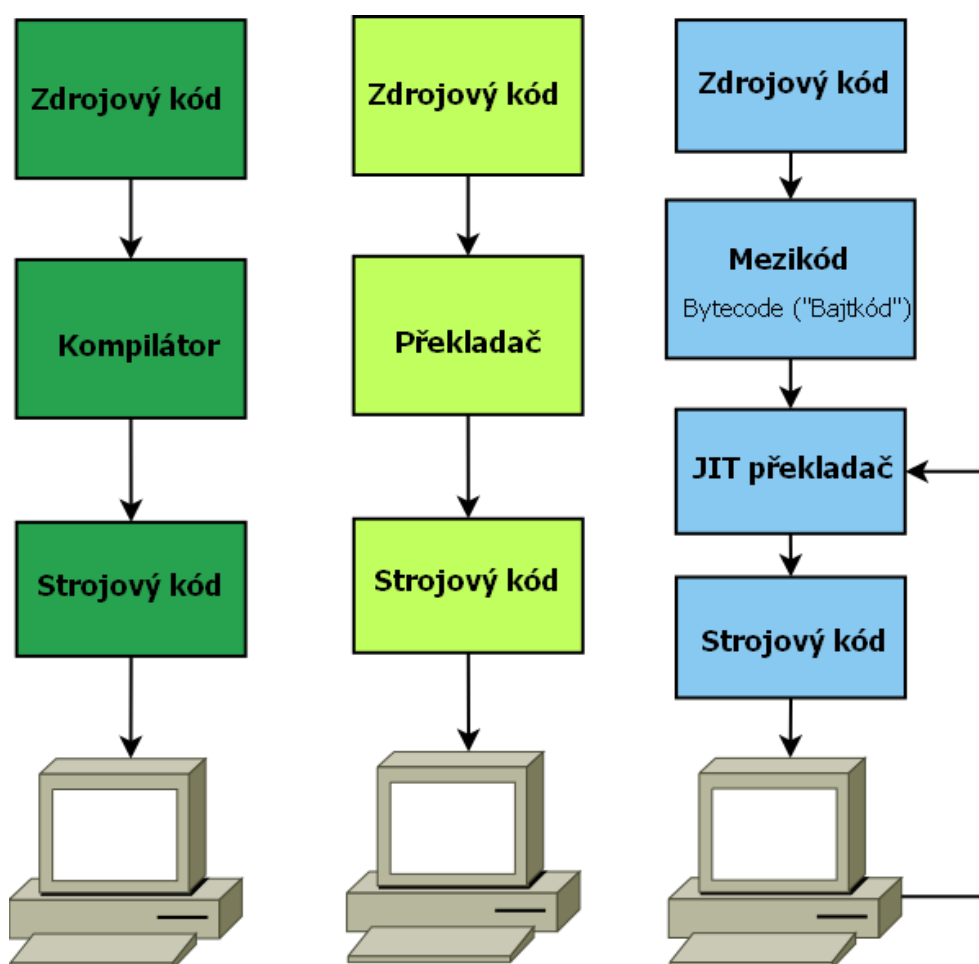
- Funguje pouze na MS Windows
- Může být za jistých okolností pomalejší
- Profesionální nástroje jsou drahé
- Možnost rekompilace zdrojového kódu (Reversní inženýrství)

⁴ [http://msdn.microsoft.com/en-us/library/aa174501\(v=sql.80\).aspx](http://msdn.microsoft.com/en-us/library/aa174501(v=sql.80).aspx) , Relational database.

⁵ <http://msdn.microsoft.com/library/zw4w595w.aspx> , .NET Framework overview.

2.2.2 C# (C sharp)

C sharp je označení pro vysokoúrovňový, objektově orientovaný programovací jazyk vyvíjený společností Microsoft. Byl vyvíjen speciálně pro platformu dot NET a v současné době je ve verzi 4.0. Jedná se o hybridní jazyk z hlediska kompilace, kde zdrojový kód není přímo kompilován do strojového kódu, ale je převeden do mezikódu, který je následně na žádost interpretován (přeložen). Této technice se říká JIT – „Just in time“, v překladu tedy „Právě v čas“. Rozdíl mezi interpretovaným, kompilovaným a hybridním jazykem si můžeme znázornit na obrázku.⁶



Obrázek 1 - Možnosti kompilace

⁶ <http://msdn.microsoft.com/library/z1zx9t92>, Introduction to C# language.

První způsob (zleva) zpracování aplikace je kompilace, kde se celý překlad do strojového kódu provádí u vývojáře. Druhý způsob je interpretace, kde překlad do strojového kódu probíhá na cílovém zařízení. Poslední, hybridní metoda kombinuje obě popsané metody.

2.2.3 SharpDevelop

SharpDevelop je tzv. „Open source“ IDE vývojové prostředí pro platformu dot NET, tedy je celé zdarma a volně ke stažení. Primárně je určeno pro jazyky C# a Visual Basic. Přestože se jedná o otevřený software, disponuje SharpDevelop spoustou užitečných funkcí včetně GUI editoru a celkově nabízí výkonné a příjemné prostředí.⁷

2.2.4 Visual Studio C#

Profesionální IDE od firmy Microsoft, které poskytuje kompletní řešení pro podporu celého životního cyklu aplikací.⁸



Obrázek 2 - Moduly visual studia

Zdroj: <http://www.microsoft.com/cze/msdn/vstudio/2010/>

Skutečně špičkové řešení ve svém oboru, kterému také odpovídá pořizovací cena.

⁷ <http://www.icsharpcode.net/OpenSource/SD/Default.aspx>, Open source IDE

⁸ <http://www.microsoft.com/visualstudio/en-us/products/Overview>

2.2.5 SQL & MS SQL Server

SQL z anglického jazyka „Structured Query Language“, tedy strukturovaný dotazovací jazyk, je programovací jazyk navržený speciálně pro práci s daty v relačních databázích. Počátky jazyka SQL se datují k roku 1970 a v roce 1986 byl přijat jako standard institucí ANSI. V roce 1992 pak byla vydána další, opravena verze.

Jazyk SQL obsahuje tyto základní části:

- DDL (Data definition language) – vytváření schémat a katalogů
- SDL (Storage definition language) – Způsob ukládání tabulek
- VDL (View definition language) – vytváření pohledů
- DML (Data manipulation language) – manipulace s daty

Microsoft SQL Server

MS SQL Server je datová platforma, čili systém, ve kterém probíhá vlastní ukládání dat nebo práce s daty. Pracuje na architektuře a klient/server a existuje v několika verzích a několika distribucích. V současné době je verze 2012 ale pro tuto práci jsem vybral verzi 2008. S touto verzí jsem měl možnost již pracovat a jedná se o prověřenou verzi s dostatečným množstvím literatury.

SQL server má několik hlavních rozhraní:

- *SQL Server Management Studio* – grafické rozhraní nad celým systémem.
- *Business Intelligence Development Studio* – Uživatelské rozhraní umožňující snadnější řešení reportovacích služeb, dolování dat, integračních služeb apod.
- *SQL Configuration manager* – Grafické rozhraní pro nastavování rozsáhlé konfigurace služeb a protokolů⁹

⁹ <http://www.sqlcourse.com/intro.html>, What is SQL?

2.3 Business intelligence

V této kapitole se budeme zabývat problematikou Business Intelligence. V první podkapitole se seznámíme s nejčastějšími definicemi. Dále ukážeme firemní oblasti, ve kterých si business intelligence najde uplatnění. Třetí podkapitola bude věnována seznámením se systémy OLTP & OLAP.

2.3.1 Základní vymezení

Co je vlastně business intelligence? Tento termín má poměrně krátkou dobu existence, to bude jeden z důvodů, proč stále ještě není zavedena jednotná definice. Poprvé s tímto termínem přišel Howard J. Dresner v roce 1989. Tehdy business intelligence popsal jako „sadu konceptů a metod určených pro zkvalitnění rozhodnutí firmy“.¹⁰

Definice podle SearchCRM

Podobná definice je i na serveru SearchCRM, která zní: „BI je určitá kategorie aplikací a technologií pro sběr, skladování, analyzování a zpřístupňování dat, jejichž účelem je pomoci podnikovým uživatelům dělat lepší rozhodnutí“.¹¹

Definice podle iOLAP

Na serveru iOLAP můžeme naléznout definici pro business jako: „sběr a analýzu dat, jejímž cílem je lepší porozumění a reakce na změny, kterým organizace čelí“.¹²

Definice podle České společnosti pro systémovou integraci

Svou vlastní definici zavedla i naše domácí společnost pro systémovou integraci. Popisuje business intelligence jako: „sadu procesů, aplikací a technologií, jejichž cílem je účinně a účelně podporovat rozhodovací procesy ve firmě“.¹³

Jak je možné vidět, všechny výše uvedené definice jsou si navzájem velice podobné. Kdybychom tedy chtěli shrnout uvedené definice, můžeme business intelligence chápat jako veškeré prostředky (software, techniky, postupy apod.) sloužící jako opora

¹⁰ http://www.computerworld.com/s/article/266298/BI_at_age_17

¹¹ <http://searchdatamanagement.techtarget.com/definition/business-intelligence>, BI

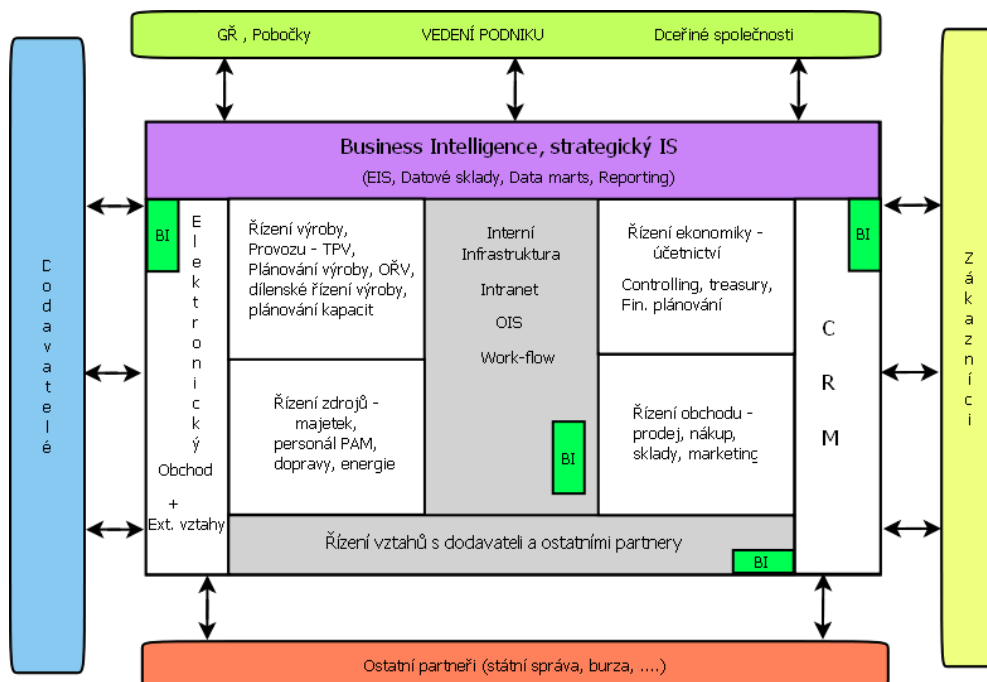
¹² <http://www.iolap.com/resources/bi-terminology>, Business intelligence

¹³ SYSTÉMOVÁ 24 INTEGRACE 2/2004, co je BI?

ke správným podnikovým rozhodnutím. Bude se tedy jednat o rozhodnutí, která budou podložena určitými daty, resp. výsledky analýz dotyčných dat.

2.3.2 Business intelligence ve firmě

Jak jsme již uvedli v kapitole 1.1.1., současná ekonomika, je ekonomika znalostní. Tedy konkurenční prostředí je stále tvrdší, a pokud chce firma zůstat konkurenceschopná, musejí podnikoví analytici a manažeři rozhodovat pod časovým tlakem a současně jejich rozhodnutí nesou velká rizika a zodpovědnost. Je tedy pochopitelné, že chtějí mít svá rozhodnutí podložena relevantními a objektivními informacemi. Informace uložené v transakčních a informačních systémech bývají vesměs ve formě relačních databází. Už samotné relační databáze nám poskytují mnoho užitečných informací a tak umožňují lepší manažerské řízení. Nicméně mnohem detailnější analýzu je možné provádět právě pomocí business intelligence, protože umožňuje nahlédnout na nasbíraná data z jiných úhlů, tedy v jiných souvislostech. Díky tomu je možné odhalit jinak těžko pozorovatelné souvislosti a informace. Tento koncept se nazývá multi-dimenzionalita a podrobně bude popsán v další kapitole.¹⁴



Obrázek 3 - Architektura BI

¹⁴ NOVOTNÝ, Ota; POUR, Jan; SLÁNKŠÝ, David. Business Intelligence, Jak využít bohatství ve vašich datech, s 20.

Nyní se budeme soustředit hlavně na využití business intelligence, resp. na útvary, které mohou využít business intelligence v rámci jedné firmy. Nejlépe nám znázorňuje vztahy mezi jednotlivými útvary firmy a business intelligence diagram na předchozí straně.

Uprostřed toho diagramu máme znázorněnu firmu. Tato firma se skládá ze čtyř hlavních útvarů, jsou to: výrobní útvar v levém horním rohu, útvar finanční v pravém horním rohu, obchodní oddělení v pravém spodním rohu a řízení zdrojů v levém spodním rohu. Samozřejmě se firmy mohou od sebe výrazně lišit podle toho, čím se firma zabývá. Pokud bychom na tyto útvary nahlédli ze systémového hlediska, každý z nich má určitý vstup a výstup. Například výrobní útvar má na vstupu suroviny a na výstupu se tvoří hotové výrobky. Obdobně je tomu u ostatních útvarů. V každém útvaru tedy dochází k určitým procesům. Procesem může být výrobní postup, vystavení faktury nebo také přijímací pohovor.

K zachycení těchto postupů resp. transakcí slouží nástroje OLTP. Zkratka znamená on-line transaction processing. Samotnými systémy OLTP se budeme zabývat v kapitole 2.3.3. Když se podíváme znovu na diagram, můžeme si všimnout, že tyto OLTP systémy jsou vlastně specifické informační systémy. Každý firemní útvar využívá informační systém pro něj určený. V ideálním případě pak využívají jednotlivé moduly korporátního informačního systému. Například pro management styku se zákazníky je určen systém CRM, jak je možné vidět v diagramu. CRM je zkratka pro „customer relationship management“ a můžeme v něm najít veškeré informace o našich zákaznících od faktur až po statistiky.

Tedy výše uvedené informační systémy, resp. OLTP systémy slouží k zachycení a uložení dat vhodných pro další zpracování. Obdobně je to i u ostatních firemních útvarů. Dále si můžeme na diagramu všimnout, že je zde velké pole zastřešující všechny útvary, tato disciplína potom bude zpracovávat data ze všech dílčích systémů a bude provádět business intelligence napříč celou společností. Různé moduly informačních systémů využívají různé úrovně managementu, celoplošný business intelligence využívá hlavně top management, k sestavování obchodních strategií, kdežto dílčí systémy v jednotlivých útvarech pak střední management k operativním plánům.

2.3.3 OLTP & OLAP

Nyní si představíme systémy OLTP a OLAP. O systémech OLTP již bylo řečeno, že zachycují transakční operace ve firmě. Zkratka znamená „On-line Transaction Processing“, tedy zpracovávání transakčních operací v reálném čase. Pojmu transakční operace můžeme rozumět jako klíčový proces, který probíhá na daném útvaru firmy a o kterém potřebujeme zaznamenávat informace. Například mějme obchodní společnost, hlavním procesem je zde prodej různého zboží. Samotná transakce je pak výměna zboží za peníze zákazníka. V tomto případě je zaznamenání informací nezbytně nutné, kvůli pozdějšímu výpočtu daně z příjmu. Systém OLTP tedy slouží k zachycení informací, o jaké se jedná zboží, jaké bylo prodané množství, jaká byla cena, popřípadě základní informace komu jsme produkt prodali. Systém nemusí být vždy na bázi výpočetní techniky, před příchodem a masivním rozšířením počítačů se tyto informace ukládali ve formě zápisů do knihy. Dále je také vhodné podotknout, že to nejsou pouze klíčové procesy, o kterých chceme mít přehled a ukládat informace. Existují procesy, jejichž výsledky nebo atributy nemusíme předkládat, avšak znalost jejich průběhu nám může výrazně přispět ke kvalitnímu řízení firmy.

Na rozdíl od systémů OLAP, o kterých se budeme vzápětí bavit, jsou na systémy OLTP kladené jiné požadavky. Jsou to např.:

- Rychlá odezva (v reálném čase)
- Nízká míra redundance (Normalizace dat)
- Možnost úpravy dat
- Možnost zálohování

Tyto požadavky přímo odpovídají relačním databázím, se kterými jsme se seznámili v úvodu práce. Oproti tomu systémy OLAP plní odlišnou funkci. Zkratka OLAP znamená „On-line Analytical Processing“, tedy analytické zpracovávání v reálném čase. Právě požadavek na zpracování v reálném čase nám tuto problematiku značně komplikuje. Na rozdíl od systémů OLTP, systémy OLAP se využívají při vytváření budoucích strategií na základě dat nasbíraných právě z OLTP systémů. OLAP systémy

však uchovávají data jiným způsobem než OLTP systémy, jedná se o tzv. datové sklady a blíže se s nimi seznámíme v dalších kapitolách. Další z odlišností mezi těmito systémy je jejich počet. Systém OLAP resp. Datový sklad bývá pouze jeden, kdežto instance systémů OLTP může být na každé firemní pobočce. Pro přehlednost si ještě uvedeme tabulku odlišností mezi oběma systémy.¹⁵

	OLTP	OLAP
Zdroj dat	Firemní procesy	Relační databáze (OLTP systémy)
Smysl dat	Základní funkce firmy	Analýzy
Rychlost zpracování	V reálném čase	až několik hodin
Velikost	Malá až středně velká	Velká až velmi velká
Druh databáze	Relační databáze	Datový sklad

Tabulka 1 - Porovnání OLTP a OLAP

2.4 Nástroje Business intelligence

Zde se seznámíme se základními nástroji pro business intelligence. Stejně jako není jednotná definice, i zde se mohou některé zdroje lišit v preferování různých nástrojů a metod. Proto si ukážeme pouze ty nejzákladnější nástroje a metody. Nejdůležitějším prvkem jsou samotné datové sklady, ve kterých uchováváme data. Dále si popíšeme nástroje pro zajištění kvality dat, ETL nástroje, datová tržiště nebo také data-mining, což je také jedna z velice důležitých metod pro získávání podkladů pro analýzy.

2.4.1 Datové sklady (Data Warehouse)

Již název nám napovídá, že tento nástroj bude sloužit k ukládání dat, které budeme následně zpracovávat při analýze BI. Termín přímo odvozený od datových skladů je datové skladování, to jsou aktivity spojené s návrhem, implementací a využíváním datových skladů. Plnění datových skladů pak můžeme rozdělit do tří kategorií.

1. Interní data
2. Externí data
3. Osobní data

¹⁵ RAINARDI, Vincent. Building a Data Warehouse with examples in SQL server, s 14.

Interní data

Nejčastěji jsou ukládána v relačních databázích a jsou hlavním zdrojem pro další zpracovávání analýz BI. Sběr interních dat probíhá prostřednictvím transakčních systémů OLTP, které si později podrobně představíme.

Externí data

Slouží jako doplněk k interním datům. Mohou dobře posloužit jako doplňující informace k faktům, které jsme analyzovali. Může se jednat o nejrozumnější průzkumy trhu od třetích stran, hospodářské výsledky a výroční zprávy konkurenčních firem.

Osobní data

Při analýzách je dobré zahrnout i data produkovaná zaměstnanci. Může se jednat o lokální databáze (MS Acces), Excelovské soubory apod. Tyto lokální databáze mohou obsahovat informace, které mohou dobře doplňovat a obohacovat hlavní data a celkově zlepšit kvalitu prováděné analýzy.

Jak bylo řečeno na začátku podkapitoly, datové sklady slouží pro ukládání dat, které dále zpracováváme v analýzách. Nabízí se tedy otázka, v čem jsou datové sklady jiné než relační databáze. Relační databáze také slouží pro ukládání dat a používají se v OLTP systémech, proč je tedy nutné zavádět nové pojmy a technologie? Jedním z hlavních důvodů je rychlost odezvy systému. Mějme na paměti, že při provádění business intelligence analýzy probíhá zpracování nad velmi velkým množstvím dat. Pokud by tyto data byla uložena v relační databázi, musela by aplikace procházet jednotlivé tabulky v každé relační vazbě a tím by se celý proces výrazně zpomalil. Datový sklad tedy obsahuje ty samé informace jako relační databáze, ale způsob uložení těchto dat je odlišný. Především je to redundance, tedy opakování se určitých dat. Například pokud v relační databázi uchováváme data o našich zaměstnancích, dodavatelích a střediscích, určitě budeme chtít vědět geografické rozložení těchto objektů. Při použití relačních databází vytvoříme tabulku, která bude obsahovat údaj o lokaci a unikátní identifikátor. Z ostatních tabulek se pak budeme pomocí relací odkazovat na konkrétní údaje o lokaci.¹⁶

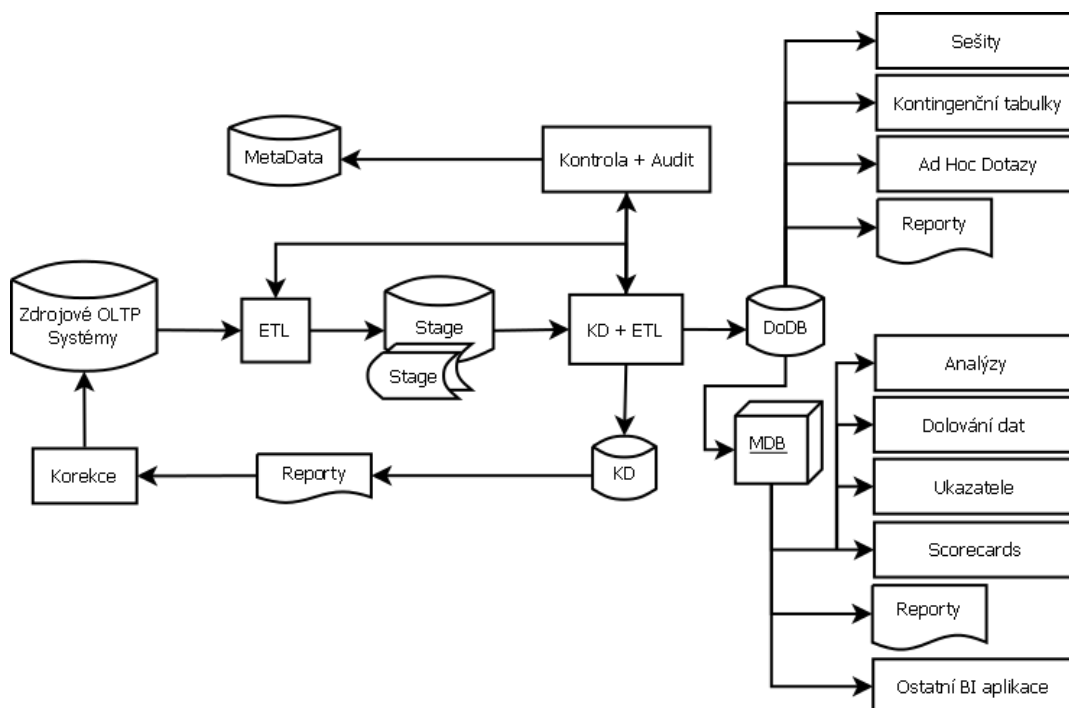
¹⁶ VERCELLIS, Carlo. Business Intelligence: Data mining and optimization for decision making, s 46.

Dále je to časové hledisko, datový sklad obsahuje data za dlouhá období na rozdíl od operačních databází. Dalším z rozdílů je neměnnost dat. Relační databáze nám umožňují data po uložení měnit (Korekce dat, aktualizace), datový sklad data pouze přijímá a není možná změna ani jejich mazání. Na závěr můžeme ještě podotknout, že datový sklad je pouze jeden a OLTP systémů může být více. V praxi to vypadá tak, že firma může mít několik poboček a na každé z nich funguje samostatný OLTP systém. Ty se pak odkazují na jeden centrální datový sklad a zásobují jej daty. Pro přehlednost můžeme uvést tabulku odlišností mezi OLTP a OLAP databázemi.

Vlastnosti	Relační databáze	Datové sklady
Čas odezvy	Řádově v sekundách	Až několik hodin
Operace	Libovolná manipulace s daty	Vkládání / čtení
Zdroje	Operační, interní	Operační, interní, externí
Velikost	Malá, střední	Velká, velmi velká
Organizace	Podle aplikace	Podle předmětu
Činnosti	OLTP	OLAP

Tabulka 2 - Porovnání relačních db a datových skladů

Nyní, když známe základní pojmy, můžeme na datový sklad nahlédnout ze systémového hlediska.¹⁷



Obrázek 4 - Architektura datového skladu

¹⁷ RAINARDI, Vincent. Building a Data Warehouse with examples in SQL server, s 10

Krok za krokem si nyní projdeme celý systém a seznámíme se s jednotlivými bloky a vysvětlíme si jejich funkci. Na levé straně začínáme u OLTP systému, které jsme si popsali v předešlých kapitolách. Tedy OLTP systémy nám poskytují data v relační formě, odtud dále pokračují k bloku ETL, kde jsou data připravena pro další zpracování. Funkci ETL si detailněji probereme v další kapitole. Po dokončení ETL procesu jsou data uložena v dočasném uložišti a vzápětí je testována jejich kvalita. Tedy blok KD reprezentuje kontrolu kvality dat. Kontrola dat podléhá auditu, který je realizován na základě metadat, což jsou informace o datech, resp. V jakém tvaru mají být nebo jakých hodnot mohou reálně nabývat. Pokud se v tomto bloku nalezne problém a data nesplní požadovanou úroveň kvality, mohou být buď automaticky opraveny na základě metadat, nebo je vygenerován report a požadavek na korekci přímo do OLTP systému.

Pokud data splňují požadovanou úroveň kvality, pak jsou zaslána do dimenzionálně orientované databáze. Dimenzionálně orientovaná databáze není nic jiného, než samotný datový sklad. V předešlých kapitolách jsme si vysvětlili, že data jsou v datovém skladu orientována podle jednotlivých dimenzí (Čas, Produkt, Zákazník apod.)

Z datového skladu pak mohou být získávány různé tabulky či konkrétní dotazy. Pro lepší analýzu jsou však data zaslána do „Multi-dimenzionální databáze“, kde získáváme nad daty zcela novou perspektivu. Datové sklady jako takové potom mohou být dále děleny, a to podle svého schéma.

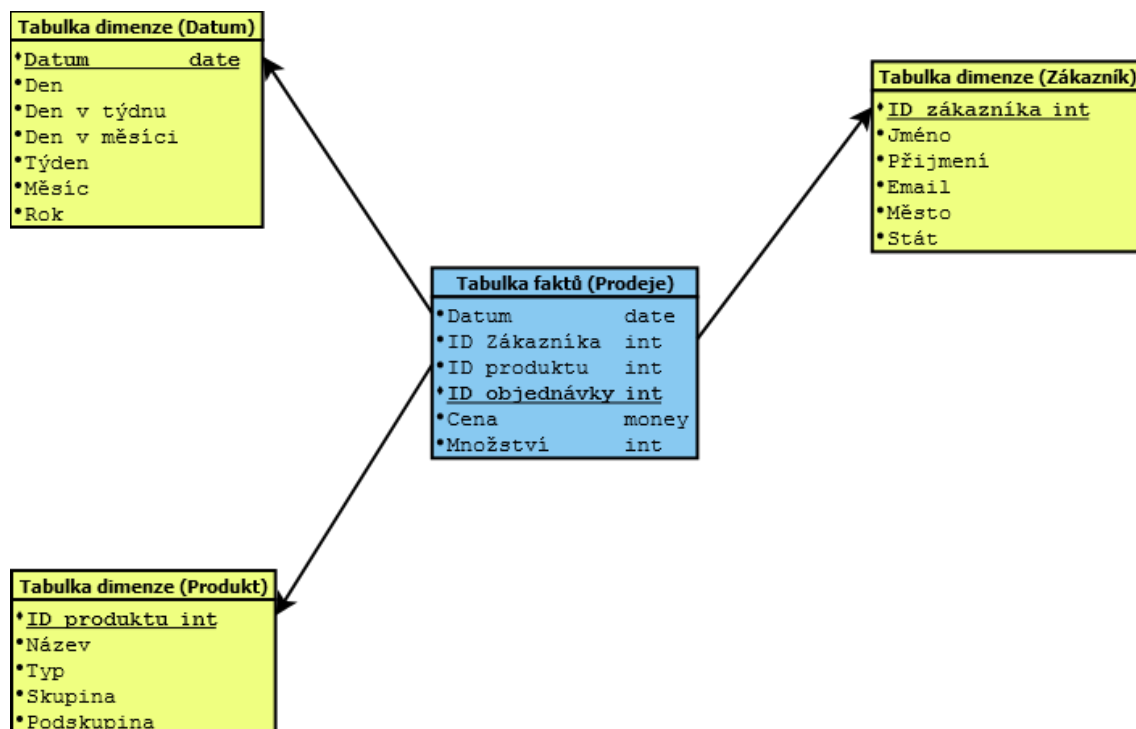
Schéma datového skladu

Z hlediska uspořádání tabulek resp. struktury datového skladu se můžeme setkat se několika různými druhy schémat. Každý druh má svoje výhody a nevýhody a každý se hodí v jiné situaci.

Schéma hvězdy

Základním druhem uspořádání je schéma hvězdy. U tohoto druhu je vysoká míra redundance, neboť zde není žádná normalizace. Najdeme zde tabulku faktů, která obsahuje pouze numerické informace, tedy mohou to být informace o prodeji, přijatém množství apod. Dále obsahuje cizí klíče, které se odkazují na tabulky dimenzí, kde se

nacházejí informace právě o dimenzích. Nejčastěji se můžeme setkat s dimenzí časovou, produktovou, geografickou, zákaznickou apod.

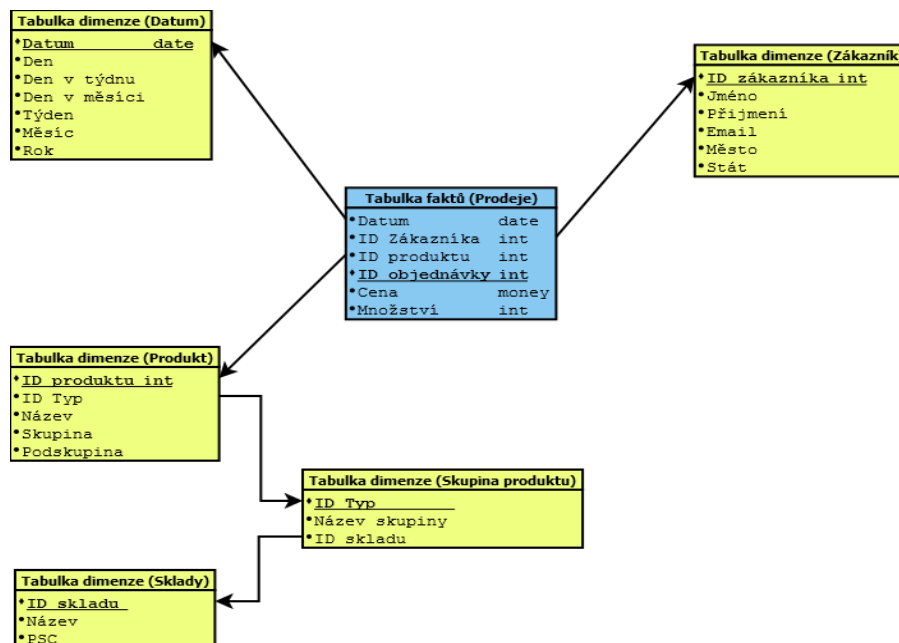


Obrázek 5 - Schéma hvězdy

Výhodou tohoto schéma je vysoký dotazovací výkon, neboť informace nemusí program dohledávat v relacích jiných tabulek. Nevýhodou je vysoká míra redundance.

Schéma sněhové vločky

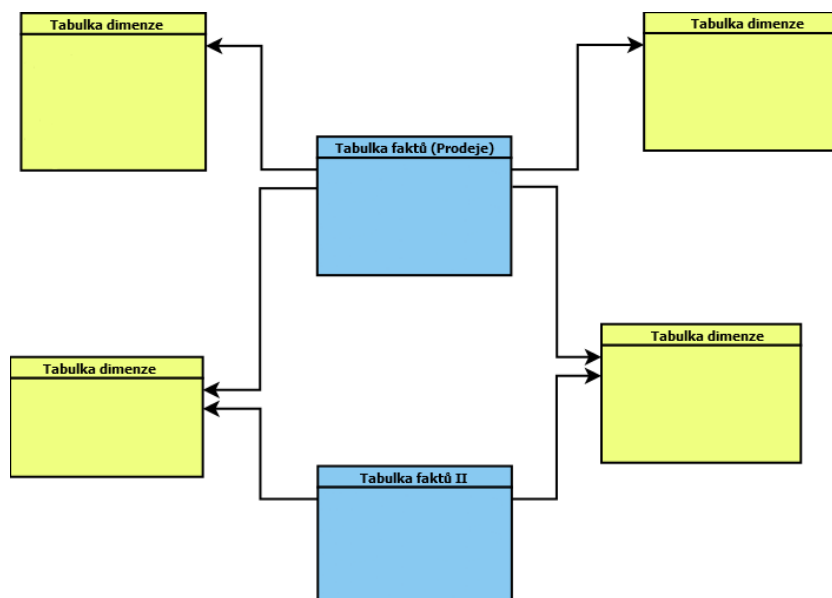
Dalším možným seskupením tabulek je „sněhová vločka“, kde se objevuje normalizace. Tedy může zde být jedna či více tabulek, které jsou relačně provázány. Toto schéma je velice podobné hvězdě, jediný rozdíl je v tabulkách dimenzí, tabulka faktů zůstává stejná. Tabulky dimenzí jsou zde řešeny tak, aby nedocházelo k redundanci. Výhodou je tedy zcela jistě nízká míra redundance a rychlejší zavádění dat. Na druhou stranou nevýhodou může být nižší dotazovací výkon, který je způsoben propojováním relačních tabulek.



Obrázek 6 - Schéma sněhové vločky

Schéma souhvězdí

Posledním možným seskupením je tzv. „Souhvězdí“, kde se nachází více než jedna tabulka faktů. Toto schéma se využívá u sofistikovaných aplikací, kde pro kvalitní analýzu potřebujeme analyzovat více tabulek faktů.¹⁸



Obrázek 7 - Schéma souhvězdí

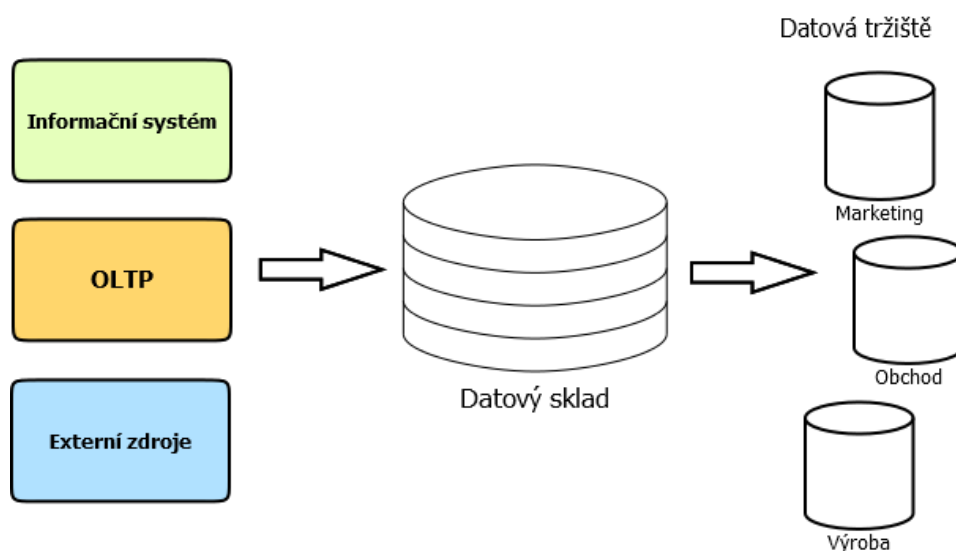
¹⁸ RAINARDI, Vincent. Building a Data Warehouse with examples in SQL server, s 71-80

Vytváření datových skladů

Pro vytváření datových skladů můžeme použít jednu ze dvou metod, metodu velkého třesku, nebo přírůstkovou metodu. V některé literatuře se můžeme také setkat s pojmy jako metoda „shora dolů“ nebo „zdola nahoru“. V této podkapitole si popíšeme, jak obě metody probíhají.

Metoda Velkého třesku

Tato metoda spočívá v konceptuálním návrhu ještě před samotnou realizací projektu. Jinými slovy si tedy nejprve detailně navrhne celou strukturu datového skladu včetně technologií, kterých budeme využívat a poté celý projekt realizujeme.¹⁹



Obrázek 8 – Metoda velkého třesku

Výhody

- Víme, co chceme, což je při řízení projektů velmi žádoucí
- Projekt může být realizován v relativně krátkém čase.

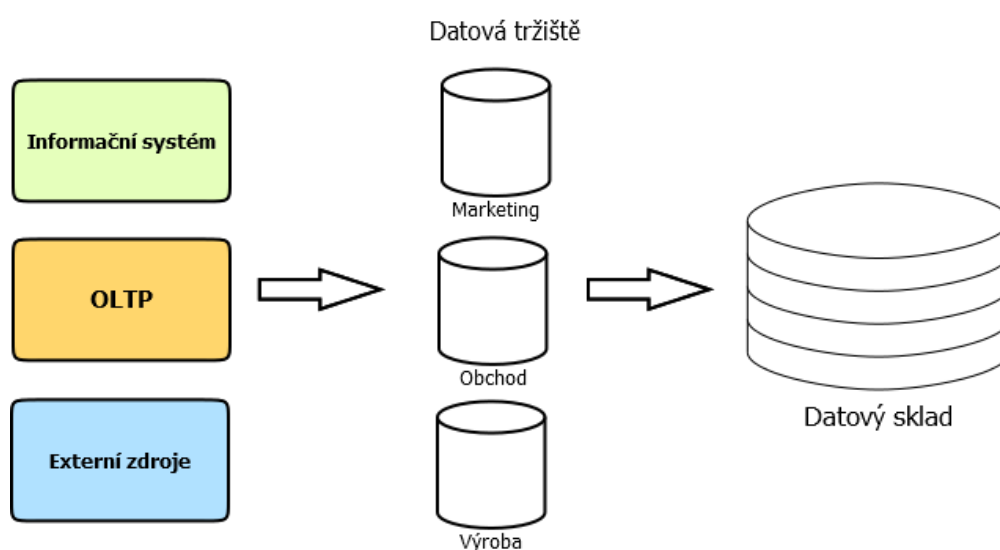
Nevýhody

- Možná změna požadavků v průběhu projektu
- Vysoké zahajovací náklady

¹⁹ VERCELLIS, Carlo. Business Intelligence: Data mining and optimization for decision making, s 49.

Metoda Přírůstková

Metoda přírůstková volí přesně opačný postup než metoda velkého třesku. Zde se počítá s průběžným a postupným budováním datového skladu na základě požadavků firmy. Čili probíhají zde jednotlivé etapy a postupně přibývají řešení, od toho je odvozen název přírůstková metoda. V praxi to pak vypadá tak, že pokud jistý útvar ve firmě pocítí potřebu datových skladů, vybuduje se právě část specifická pro tento útvar. Začíná se tedy od tzv. datových tržišť. Datová tržiště si popíšeme v následující kapitole.²⁰



Obrázek 9 – Metoda přírůstková

Výhody

- Nízké zahajovací náklady
- Možnost reakce změny na požadavky
- Rychlejší návratnost investic

Nevýhody

- Vybudování kompletního datového skladu může trvat dlouho
- Problematictější promítnutí strategických rozhodnutí

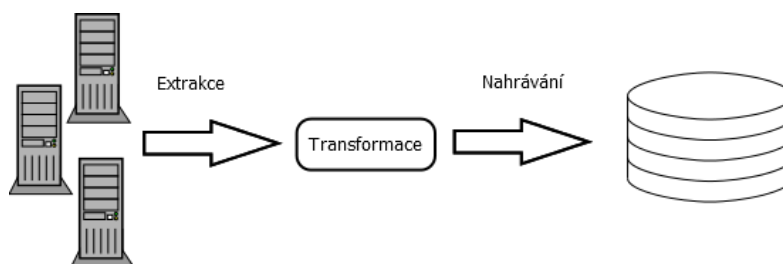
²⁰ VERCELLIS, Carlo. Business Intelligence: Data mining and optimization for decision making, s 49.

2.4.2 Datová tržiště (Data Mart)

Datová tržiště jsou systémy, které shromažďují data požadované určitým útvarem ve firmě. Jak bylo popsáno v kapitole 1.2.2, kde jsme si graficky znázornili útvary, které mohou využívat business intelligence. Každý z těchto útvarů bude potřebovat pro svoje analýzy jiná data, např. výrobní oddělení se nebude zajímat o informace o klientech firmy. Tím pádem můžeme považovat datové tržiště za funkční část datového skladu. Data využívaná datovým tržištěm jsou tedy podmnožinou celkových dat uložených v datovém skladu. V našem případě, kdy máme nashromážděná data o dodávkách zemědělských komodit do skladů firmy, se jedná o údaje, které využijí jen určité útvary firmy.²¹

2.4.3 Transformační nástroje (ETL)

Již několikrát jsem se zmínil o procesu ETL, v této kapitole se na tento proces detailně podíváme. Zkratka ETL znamená „Extraction, transformation, loading“, tedy extrakci, kdy získáváme data z různých zdrojů, transformace, kde jsou data transformována do jednotného formátu pro datový sklad a zároveň je zajištěna určitá kvalita dat a nakonec loading, tedy samotné nahrávání do datového skladu. Blokové schéma je tím pádem velice jednoduché a skládá se ze tří částí, jak je uvedeno výše.



Obrázek 10 - ETL proces

Extrakce

Teoreticky by mohl být celý tento proces vnímán jako jednoduchý, v praxi však může být velice náročný. Vezmeme-li první krok extrakci, již zde může nastat problém. Jak bylo popsáno v kapitole o datových skladech, jejich zdroji mohou být databáze nebo

²¹ NOVOTNÝ, Ota; POUR, Jan; SLÁNKŠÝ, David. Business Intelligence, Jak využít bohatství ve vašich datech, s 33.

různé datové soubory, soubory tabulkových procesorů apod. Tím pádem nám tedy vždy nemusí stačit jediná aplikace, resp. jediné řešení pro extrakci.

Dalším aspektem je množství dat, které chceme extrahovat. Pokud zavádíme datový sklad společně s OLTP systémy, pak se budou data extrahovat postupně a nehrozí zatížení transakčních systémů. V praxi však bývá daleko častější zavádění datových skladů později, tedy extrakce dat zpětně za několik let. Takové množství dat může znamenat i několik Gigabytů přenosu, což nelze provádět naráz, aniž bychom nevyřadili OLTP systém z provozu. V takových situacích se volí postupné zavádění dat v době, kdy nevedí zpomalení OLTP systémů, nebo po malých částech tak, aby systémy nebyly zatíženy příliš.

Vezmeme-li proces ETL jako třífázový, pak extrakci obstarává tzv. „datová pumpa“. Jak bylo již řečeno, tato část obstarává získávání dat ze zdrojových systémů. To je možné provádět několika možnými způsoby, podle toho, který systém si vyžádá extrakci.

Způsob „Pull“

Z anglického „pull“ tedy „vytáhnout“ je to cílový systém, který žádá o data zdrojový systém. Dalo by se tedy říct, že data „vytahuje“. Při tomto způsobu může docházet k vyčerpání datového serveru natolik, že je zpomalená jeho reakce. Tento způsob se využije při zpětném získávání dat

Způsob „Push“

Z anglického „push“ tedy „tlačit“ je to naopak zdrojový systém, který pomocí tzv. triggerů (spouští) posílá data do cílového systému. Spoušť je na SQL serveru spuštěna při určité události, např. aktualizace tabulky, přidání záznamu nebo mazání záznamu. Extrakce tedy probíhá průběžně.

Pomocí logů

Zdrojový systém může data zaznamenávat do tzv. logů, ze kterých si pak může cílový systém data čerpat. Během čerpání dat z logů není zdrojový systém zatěžován. Logy jsou postupem času odmazávány, aby nedocházelo ke zbytečnému využívání prostoru.

Manuální extrakce na požádání

Data ze zdrojového systému můžou být také extrahována na požádání manuálně.

Transformace

Poté, co máme data vyextrahována, je potřebujeme ještě transformovat do jednotného formátu, který využívá datový sklad. Opět je zde několik možností jak postupovat.

- Transformace s mezi uložením
- Transformace v paměti
- Zpětná transformace (ELT)

Transformace s mezi uložením

Pokud zacházíme s velkým obsahem dat, je nutné tyto data před samotnou transformací uložit do dočasného uložistiště. Transformace probíhá poté.

Transformace v paměti

Pokud to velikost a výkon ETL serveru umožňují, můžeme data transformovat přímo v operační paměti serveru. Tento způsob je rychlejší než způsob s ukládáním.

Zpětná transformace

Tento způsob můžeme využít, máme-li silné servery s paralelním zpracováním. Data jsou nejprve nahrána do datového skladu a teprve až potom jsou transformována, proto označení ELT („extraction-loading-transformation“).

Nejčastější problémy při transformaci:

- Formát data (času DD-MM-RRRR / MM-DD-RRRR)
- Obecně formát záznamů (např. rozdílný zápis PSČ „39601“ a „396 01“)
- Rozdílná měna
- Rozdílné názvy atributů (např. klient/zákazník)
- Duplicita dat
- Neúplná data

Nahrávání

Poté, co jsou data vyextrahována a transformována do požadované podoby a kvality, jsou nahrány do datového skladu.²²

2.4.4 Reporting

V momentě, kdy máme datový sklad správně navržený, transakční systémy nám sbírají data a ETL procesy nám plní datové sklady kvalitními daty, pak už nezbývá nic jiného než data zpracovat a výsledky prezentovat. Jednou z možných metod prezentace je právě reporting. V předchozích kapitolách to ale byl systém OLAP, o kterém jsem tvrdil, že zpracovává data z datových skladů a poskytuje manažerům podklady pro správná rozhodnutí. Jaký je tedy rozdíl mezi OLAP a reportingem? Tedy obě metody, reporting i OLAP zpracovávají data z datového skladu. Hlavním rozdílem je způsob vytváření, OLAP analýza je interaktivní proces, kde uživatel zadává různé požadavky a systém na ně reaguje. Oproti tomu report může být generovaný systémem automaticky bez přítomnosti manažera nebo pracovníka. Reporty nemusejí nutně obsahovat pouze výsledky dat, které shromažďuje datový sklad (např. přehled prodejů, poruchovosti apod.), reporty mohou obsahovat i informace o provozu celého skladu a poskytují tak dobré podklady pro údržbu a bezproblémový provoz. Reporty tedy můžeme dělit do několika kategorií:

- Reporty o kvalitě dat
- Reporty o využití (kdo, kdy, kde, co jak vyžadoval nebo používal)
- Jednodimenzionální reporty (Obsahují fakta vzhledem jedné dimenzi)

Výhodou reportů je tedy jednoznačně to, že k jejich vytváření není potřeba kvalifikovaného pracovníka. Tyto reporty pak mohou poskytovat základní informace i nezkušeným pracovníkům, nebo těm, kteří neovládají práci s OLAP systémem. Na druhou stranu report se kvalitou poskytnutých informací nemůže rovna detailní OLAP analýze.²³

²² SILVERS, Fon. *Building and Maintaining a Data Warehouse*, s 155.

²³ RAINARDI, Vincent. *Building a Data Warehouse with examples in SQL server*, s 329

2.4.5 Multi-dimenzionální databáze

V předchozí kapitole jsem popisoval prezentaci dat pomocí reportingu, a nyní se podíváme na druhý způsob, OLAP analýzu. Aby mohla být vůbec realizována OLAP analýza, musí existovat multi-dimenzionální databáze. Možnost pohlédnout na data z více hledisek (dimenzí) jsem na začátku práce popisoval jako klíčovou výhodu business intelligence.

Multi-dimenzionální databáze je druh databáze, kde jsou data uložena jiným způsobem, než v relační databázi. Data jsou uložena v buňkách a pozice každé buňky je definována několika dimenzemi. Jinými slovy, celou databázi si můžeme představit jako trojrozměrnou kostku složenou z malých kostek (buněk), kde každá hodnota v buňce představuje obchodní událost. Samotnou datovou kostku si probereme zvlášť v následující kapitole, nyní se budeme soustředit na jednotlivé metody stavby multi-dimenzionálních databází a jejich vlastností. Fyzicky je databáze realizována souborem a data jsou ještě komprimována.

Při multi-dimenzionální OLAP analýze můžeme využívat jeden z těchto způsobů uložení multi-dimenzionálních dat:

- MOLAP (Multi-dimenzionální OLAP)
- ROLAP (Relační OLAP)
- HOLAP (Hybridní OLAP)
- DOLAP (Desktop OLAP)

MOLAP

Při této metodě jsou data uložena v kostce, kostka vyžaduje prostor k uložení a celkově je výpočetně náročnější, navíc zde vzniká redundance, protože jsou stejná data uložena v datovém skladu a v multi-dimenzionální databázi. Na druhou stranu tato metoda poskytuje kvalitní výsledky.

ROLAP

Tato metoda nevytváří novou databázi, ale čerpá data přímo z relačních vazeb datového skladu. Tím pádem tedy nevzniká redundance, ale výsledná kvalita analýzy není tak vysoká jako v případě MOLAP metody.

HOLAP

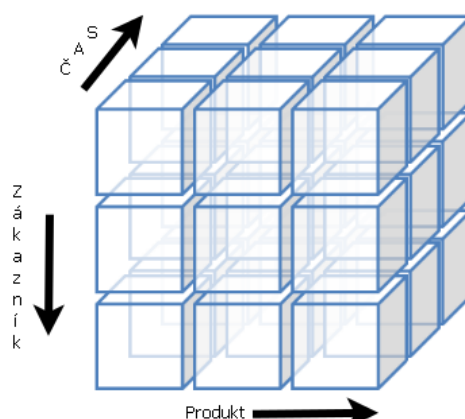
Hybridní metoda kombinuje obě výše uvedené metody a využívá jejich silných stránek. Do multi-dimenzionální databáze jsou ukládána pouze některá data a zbylé potřebné údaje jsou získány přes relační vazby. Tato metoda nabízí kompromis mezi výkonovými nároky a kvalitou výsledné analýzy.

DOLAP

Poslední možná metoda data ukládá na cílové stanici uživatele, který provádí analýzu. Z toho pochází název „Desktop“. Nároky na zpracování se tím pádem přesouvají na koncové stanice.²⁴

2.4.6 OLAP Datová kostka

Datová kostka je výsledkem právě multi-dimenzionální databáze. Data, která jsme nahrávali do datového skladu a pak uložili do multi-dimenzionální databáze, nyní analyzujeme pomocí datové kostky. Pro lepší představu je datová kostka na obrázku. Každá hodnota v buňce představuje obchodní událost, kde pozice v buňce určuje její dimenze. Např. Pohyb po ose X by nám měnil produkt a pohyb po ose Y by nám měnil zákazníka.



Obrázek 11- Datová kostka

²⁴ SILVERS, Fon. *Building and Maintaining a Data Warehouse*, s 210.

Základní operace s datovou kostkou

Abychom se dopracovali k požadovaným výsledkům, musíme s nově vytvořenou datovou kostkou provádět nejrůznější operace. Samotná datová kostka nám informace nepřináší, je to právě výše zmíněná interaktivní analýza.

Možné operace s datovou kostkou:

- Rotace
- Slice & dice
- Drill down
- Roll up
- Drill accross

Rotace

Rotace patří mezi ty nejjednodušší operace, které můžeme s datovou kostkou provádět. Datovou kostku můžeme tedy libovolně otáčet a měnit tak náš pohled na dimenze.

Slice & dice

Tato operace nám umožňuje nahlédnout na data „uvnitř“ kostky. Lépe řečeno chceme-li nahlédnout na data v kostce vzhledem k jedné z dimenzí, pak nám tato operace umožní „ukrojit“ požadovanou část kostky a na data nahlédnout.

Drill down

Operace drill down nám umožní „zanoření“ do hierarchie určité dimenze. Například, pokud máme v kostce data o prodeích určitých produktů v čase a regionu, pak nám tato operace umožní přejít z časových hodnot roků na kvartály, či měsíce apod. Samozřejmě pouze za předpokladu, že jsou data uložena v takovém formátu, který to umožňuje.

Roll up

Operace roll up umožňuje sumační operace. Máme-li uloženy prodeje za měsíce, sumační operace přepočítá tyto údaje na kvartály jednotlivých let. Jedná se tedy o přesně opačný proces, než při operaci drill down.²⁵

²⁵ RAINARDI, Vincent. Building a Data Warehouse with examples in SQL server, s 413.

2.4.7 Data-mining

Z anglického slova „mining“ – dolování, můžeme tento pojem přeložit jako dolování nových dat z těch stávajících. Opět bychom se zde mohli zastavit a namítat, že reporting a OLAP analýza nám také poskytují informace z uložených dat. Mezi Data-miningem a OLAP analýzou je však zásadní rozdíl a to z časového hlediska. Při OLAP analýzách sledujeme události a veličiny, které již nastaly, kdežto u data-miningu se snažíme najít pravidelnost či závislost a s pomocí toho, data predikovat. Máme-li tedy údaje z předchozích časových období, můžeme pomocí vhodného algoritmu provést predikci budoucích hodnot námi sledovaných veličin.

Algoritmy:

- Rozhodovací stromy
- Shlukování
- Asociační pravidla
- Časové řady
- Neuronové sítě

Rozhodovací stromy

Tento algoritmus patří mezi velice používané a oblíbené techniky dolování dat. Algoritmu je intuitivní a dobře pochopitelný. Cílem tohoto algoritmu je rozdělit určité entity do skupin podle nějakého atributu (např. požitelnost u hub, cena u auta apod.). Parametry podle kterých se algoritmus rozhoduje, jsou atributy daných entit. Předtím, než můžeme algoritmus použít, potřebujeme množinu entit s atributy s již definovanou vlastností, kterou sledujeme. Tato množina slouží jako učící vzorek, na které se algoritmus naučí rozhodovat. Tento algoritmus je velice rychlý a může poskytovat kvalitní analýzy, nicméně ne do všech situací se hodí.

Shlukování

Shlukování je algoritmus, který se používá na odhalování shluků dat, tedy skupin entit, které jsou si určitým způsobem podobné. Kritéria pro tento algoritmus nejsou předem zadána, odhalují se přirozené struktury – shluky dat.

Asociační pravidla

Asociační pravidla jsou rovněž určena pro odhalování vztahů v datech. V tomto případě však není žádný atribut vyčleněn jako cíl klasifikace. Asociační pravidla zkoumají všechny závislosti. Z hlediska statistiky se jedná o hledání korelace.

Časové řady

Časové řady se hodí pro predikci proměnné v čase. Máme-li tedy data o proměnné z minulosti a současnosti, je možné pomocí tohoto algoritmu předvídat budoucí trend proměnné. Tento způsob analýzy v sobě zahrnuje i cyklické fluktuace, tedy pravidelné kolísání hodnot.

Neuronové sítě

Neuronové sítě funkčně vycházejí z lidského mozku, stejně jako tam, je systém složen z množiny navzájem propojených elementů – neuronů. Tyto neurony na sebe navzájem působí a tím vznikají podněty pro určité chování. Stejně jako u některých jiných algoritmů, je zde potřebný vzorek dat pro učení. Poté se algoritmus provádí v cyklech (iteracích) dokud není dosažena požadovaná přesnost výsledku.²⁶

²⁶ CAO, Longbing; YU, Philip; ZHANG, Cheng; ZHANG, Huaifeng. Data mining for Business Applications, s 217.

3. ANALÝZA PROBLÉMU A SOUČASNÉ SITUACE

3.1 Charakteristika společnosti

Základní údaje o společnosti

Obchodní název společnosti: ZZN Pelhřimov a.s.

Sídlo společnosti: Pelhřimov, Nádražní 805, PSČ: 393 57

IČ společnosti: 46 67 81 40

Právní forma společnosti: akciová společnost

Založení a vznik společnosti:

Společnost vznikla dnem 1.5.1992. zápisem do obchodního rejstříku a od tohoto dne je zapsána v obchodním rejstříku vedeném Krajským soudem v Č. Budějovicích, oddíl B, vložka 496.

Předmět podnikání společnosti:

- Koupě zboží za účelem jeho dalšího prodeje a prodej kromě zboží vyžadujícího zvláštní povolení
- Výroba krmných směsí, krmných koncentrátů, minerálních přísad, doplňků biofaktorů a surovin pro jejich výrobu
- Výroba a uvádění do oběhu uznaného osiva a sadby polních plodin
- Služby pro zemědělskou výrobu
- Zemědělská výroba

*„Akciová společnost ZZN Pelhřimov patří mezi nejvýznamnější obchodní společnosti působící v regionu Českomoravské Vysočiny a Jižních Čech. Výroba a prodej jednotlivých produktů je certifikována v rámci integrovaného managementu systému mezinárodní společnosti Det Norske Veritas. V rámci tohoto systému je certifikováno plnění požadavků norem ISO 9001:2000, ISO 14001:2004, OHSAS 18001:1999, HACCP, QS Systém a GTP“.*²⁷

²⁷ <http://www.zznpe.cz>, Certifikáty

Organizační struktura společnosti

Společnost ZZN Pelhřimov a.s. je členem skupiny AGROFERT a.s.

Společnost AGROFERT a.s. vlastní významné majetkové účasti ve zpracovatelských, výrobních a distribučních podnicích zemědělského, potravinářského a chemického průmyslu. [5]

Chemie	Zemědělství	Potravinářství	Pozemní technika
Precolor	Agroservis Tachov	Kostecké uzeniny	AGROTEC
CS Cabot	Navos	Maso Planá	AGRI CS
Deza	Zenza Znojmo	PENAM, a.s.	
DUSLO a.s.	ZZN Pardubice	HYZA, a.s.	
Duslo, a.s. OZ ISTROCHEM	ZZN Havlíčkův Brod		
FATRA, a.s.	ZZN Pelhřimov		
Kemifloc	ZZN Pomoraví		
Lovochemie	ZZN Rakovník		
Precheza	ZZN v Mělníku		
SKW Piesteritz	Lipra		
Synthesia a.s.	ZZN Polabí		
	ZENA Mladá Bol.		

Tabulka 3: Seznam členů skupiny AGROFERT a.s.

3.2 Produkty firmy

Firma má široké portfolio produktů a služeb. Produkty můžeme rozdělit do několika hlavních kategorií.

- Krmné směsi
- Minerální hnojiva
- Osiva
- Paliva
- Agro služby

Krmné směsi

Firma vyrábí krmné směsi pro všechny kategorie běžně chované zvěře (Prasata, drůbež, Skot, atd.). Kapacita výroby se pohybuje kolem 200 000 tun ročně.

Firma také nabízí službu, kde si zákazník může nechat namíchat krmnou směs z vlastních surovin a přímo v místě, které mu vyhovuje. Jako jedna z mála firem, firma vlastní certifikát pro výrobu medikovaných krmných směsí. Samozřejmě všechny směsi prochází přísnou kontrolu kvality a hygieny.

Minerální hnojiva

Firma dále prodává kompletní sortiment zemědělských hnojiv. V základní nabídce jsou Síran amonný krystalický, Ledek amonný s dolomitem, DASA 26-13, Močovina, DAM 390, Amofos NP 12-52. Firma je také schopna vyrobit hnojivo přímo dle požadavků zákazníka.

Osiva

Další položkou portfolia firmy jsou osiva. Firma disponuje několika čistícím stanicemi a také využívá moderní mořící techniky. K dostání jsou základní druhy obilovin, luskovin, jetelovin a ostatních plodin.

Paliva

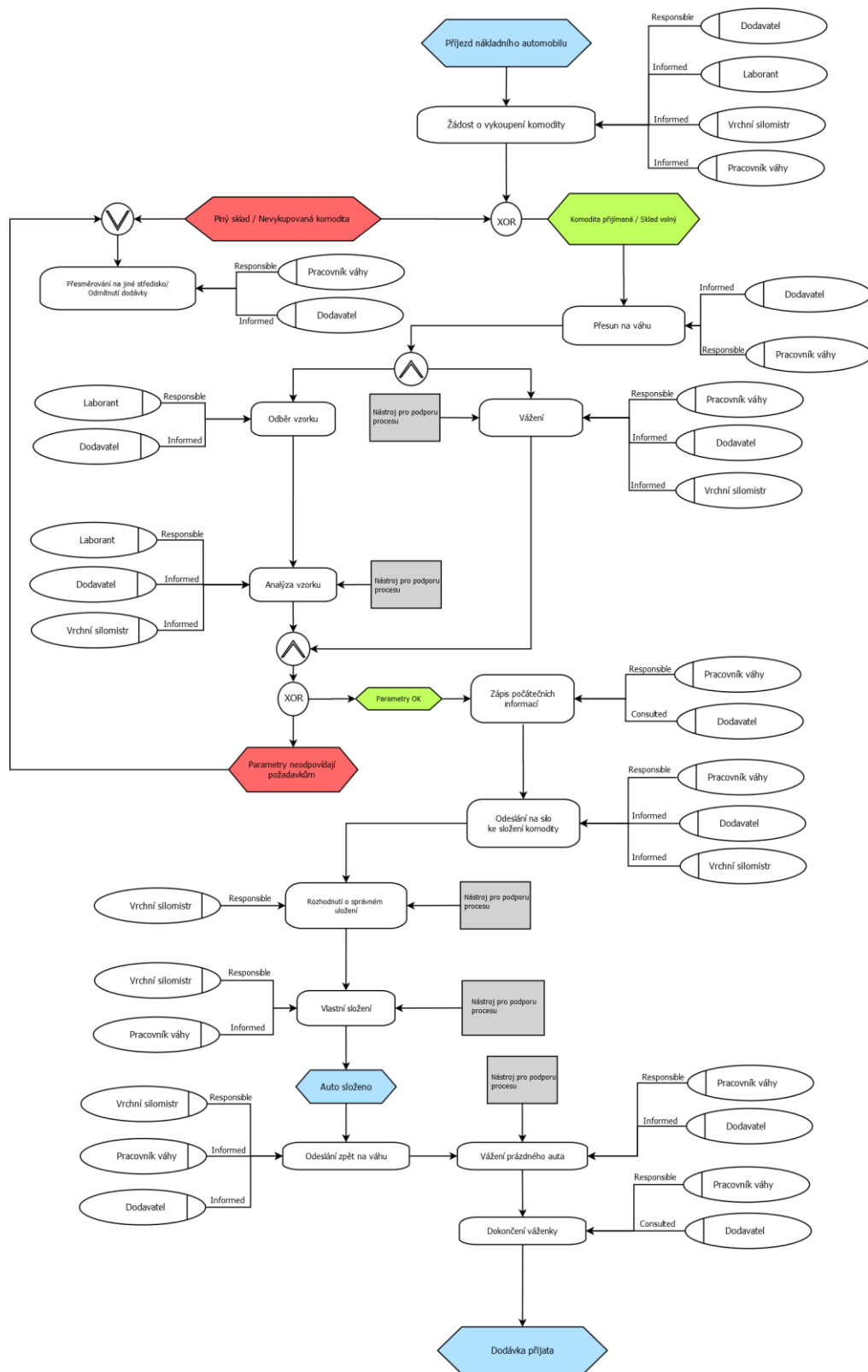
Pro své zákazníky firma také nabízí motorovou naftu a uhlí.

Agro služby

Nejmladší položkou portfolia jsou právě agroslužby. Nabízeny jsou od roku 1995 a zákazníci je rádi využívají.

V této práci se budu soustředit na divizi, která má za úkol nákup zemědělských komodit a jejich další uskladnění. Tyto komodity se pak dále prodávají nebo jsou použity na výrobu zmíněných krmných směsí. Hlavním procesem v této divizi je výkup zboží od zemědělců. Pro lepší představu si uvedeme EPC diagram.

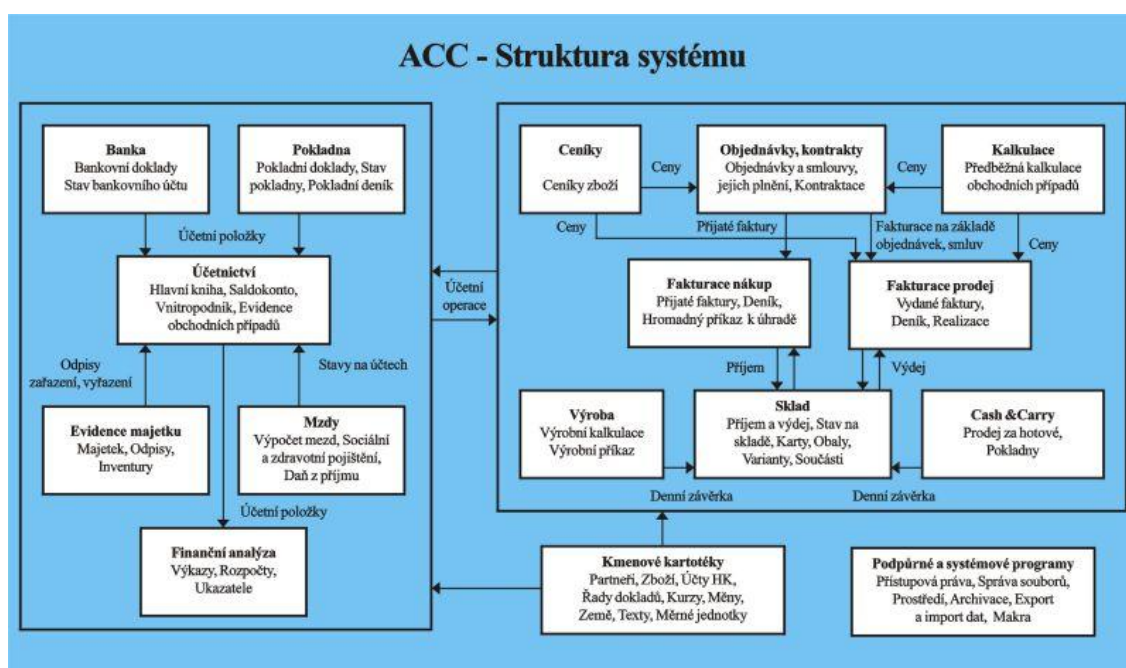
3.3 Hlavní proces – EPC diagram



Obrázek 12 - EPC diagram hlavního procesu

3.4 Současné využití BI ve společnosti

V podniku je zaveden informační systém od firmy Accord Software-House – ACC. Jedná se o modulový ekonomický systém pro řízení podniku. V tomto informačním systému je možno si sestavit moduly přímo podle individuálních potřeb každé firmy. Program je napsán v jazyce C a data jsou uložena ve formátu dBASE. Jednotlivé firemní útvary se připojují pomocí technologie VPN a programu vzdálená plocha. Poté spouštějí individuální moduly pro svou práci. Oba systémy OLTP a OLAP zde tedy zastupuje již zmíněný informační systém.



Obrázek 13 - Struktura ACC informačního systému

Pro zpracovávání analýz má pak firma vyhrazeného jednoho zaměstnance – analytika. Bavíme-li se tedy o business intelligence, probíhá na nejvyšší úrovni podle obr. 1 a to v top managementu. Ke zpracování se využívá stejný informační systém, který umožňuje vytváření různých sestav a statistik, nicméně zde chybí prvek multi-dimenzionality a je vhodné navrhnout nástroj pro takové zpracování. Jednotlivé útvary tedy nezpracovávají žádné analýzy, pouze sbírají transakční data. Tím pádem se domnívám, že analýzy na nižších úrovních, tedy přímo na jednotlivých střediscích a

v rámci jednoho útvaru, by mohly odhalit určité informace a spojitosti, podle kterých by bylo možné zkvalitnit řízení a zlepšit tak situaci firmy.

Firma chce zároveň do pololetí příštího roku zavést nový informační systém. Konkrétně je to systém od společnosti SAP. Hlavní důvody pro tento systém jsou:

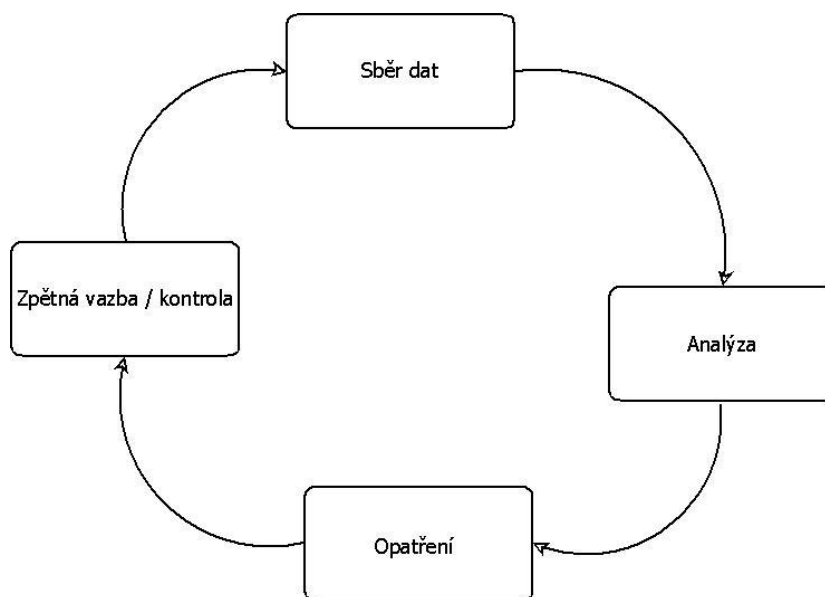
- Garance funkčnosti od výrobce spojena s podporou
- Pozitivní reference
- Vysoká integrace
- Podpora nových OS
- Podpora Business intelligence

3.5 Návrh řešení

Vlastní analýza bude probíhat v následujících krocích:

1. Sběr dat – Export z ACC systému nebo ruční zadání do vytvořeného programu
2. ETL fáze
3. Uložení dat do datového skladu
4. Vytvoření multi-dimenzionálního modelu (Datové kostky)
5. Příslušné operace s datovou kostkou (Slice & dice, Drill down/up, rotace apod.)
6. Grafická interpretace
7. Slovní interpretace vydolovaných informací
8. Návrhy na zlepšení
9. Nový cyklus.

Jak je vidět jedná se neustálý koloběh procesů, tedy nikoliv jednorázová analýza, která má přinést zlepšení. Po zavedení návrhu se samozřejmě musí sledovat dopady a na základě toho provést hodnocení. Čerstvě nasbíraná data nám tak poskytnou opět nové informace a poznatky. Jedná se v podstatě o Demingův model PDCA, kde se neustále opakuje cyklus plánování (plan), realizace (do), kontrolování (check) a jednání (act). Tím se zajišťuje neustálé zlepšování služeb a výrobků.

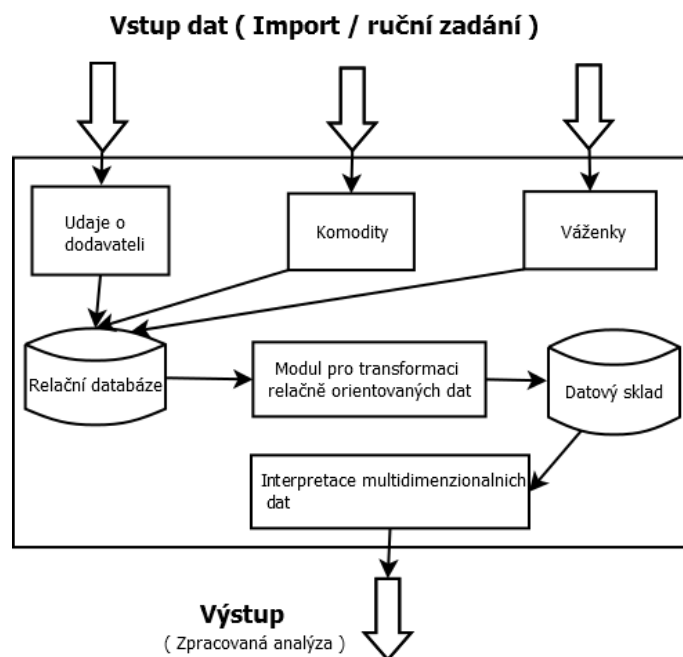


Obrázek 14 - Proces provádění analýz

V našem případě máme k dispozici transakční data z několika středisek. Střediska se mezi sebou liší jak geograficky, tak složením dodavatelů nebo technickou vybaveností. Samozřejmě personál je také odlišný a kromě podnikových směrnic zde není velký prostor pro sdílení informací.

Návrh softwarové podpory

Softwarový nástroj, pomocí kterého budeme analýzu zpracovávat, můžeme popsat ze systémového hlediska tímto způsobem: Jako vstupy bude program přijímat informace o dodavatelích, komoditách a pak jednotlivé váženky, což jsou transakční data z jednotlivých středisek. Jednotlivé moduly pro konkrétní vstupy budou přijímat data a ukládat je do relační databáze. Každý ze vstupních modulů bude mít zpětnou vazbu, která se bude starat o správnost vkládaných dat. Po nashromáždění dostatečného objemu dat, můžeme přistoupit k transformaci relačních dat do datového skladu. Po naplnění datového skladu můžeme vytvořit datovou kostku a různě s ní operovat. Poté přichází na řadu poslední modul, který má za úkol interpretovat analyzovaná data formou grafů a kontingenčních tabulek. Na základě interpretovaných informací bude možné navrhnout postupy pro zlepšení situace. Pro lepší představu nám poslouží následující diagram.



Obrázek 15 - Blokové schéma softwarové podpory

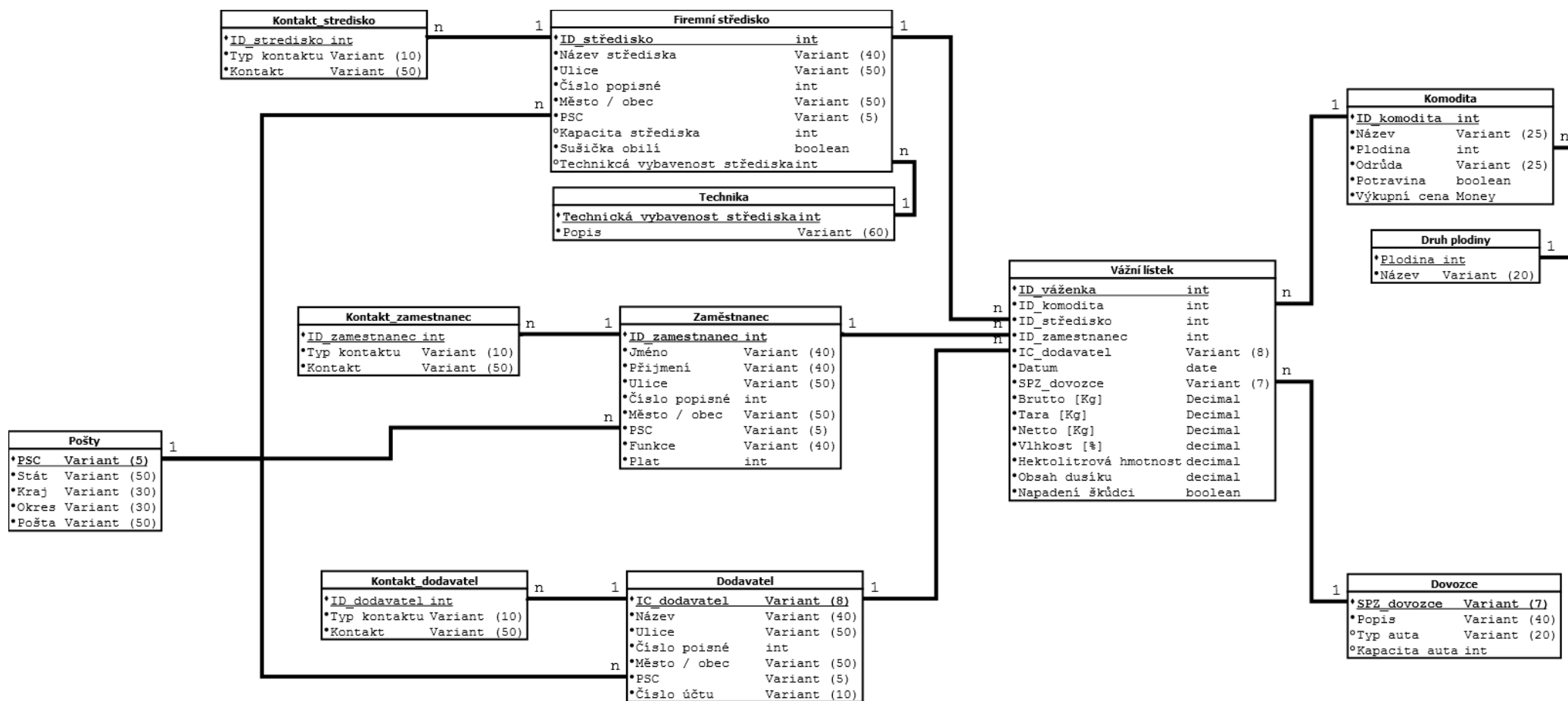
Možné analýzy:

- Z geografického hlediska analyzovat kvalitu přijímaných komodit a efektivitu skladování. Vyhodnotit nejúspěšnější střediska a pokusit se zaznamenat klíčové postupy a vlivy do znalostní báze pro široké užití.
- Z geografického hlediska analyzovat výskyt škůdců/nemocí na komoditách. Vyhodnotit preventivní opatření nebo efektivní nápravná opatření.
- Z geografického hlediska analyzovat druhy pěstěných komodit a na základě toho Vyhodnotit, kterým dodavatelům by bylo vhodné nabídnout firemní služby a konkrétní druhy hnojiv (+efektivní skladování).
- Z dodavatelského hlediska analyzovat kvalitu komodit a způsobu dodání. Vyhodnotit, kterým dodavatelům by bylo vhodné nabídnout firemní služby (Sklízení, přeprava, pokročilé metody hnojení nebo ošetření proti škůdcům)
- Z časového hlediska analyzovat výskyt škůdců/nemocí na komoditách. Vyhodnotit preventivní opatření nebo efektivní nápravná opatření (Odlišné vlivy prostředí).
- Podle technické vybavenosti jednotlivých středisek analyzovat úspěšnost skladování a vyhodnotit možné investice.

- Z produktového hlediska (Komodity) analyzovat geografické rozložení a efektivně tak stanovit možné přepravy a zásobování.
- Z produktového hlediska (Komodity) analyzovat možnosti obchodování na zahraničních trzích.

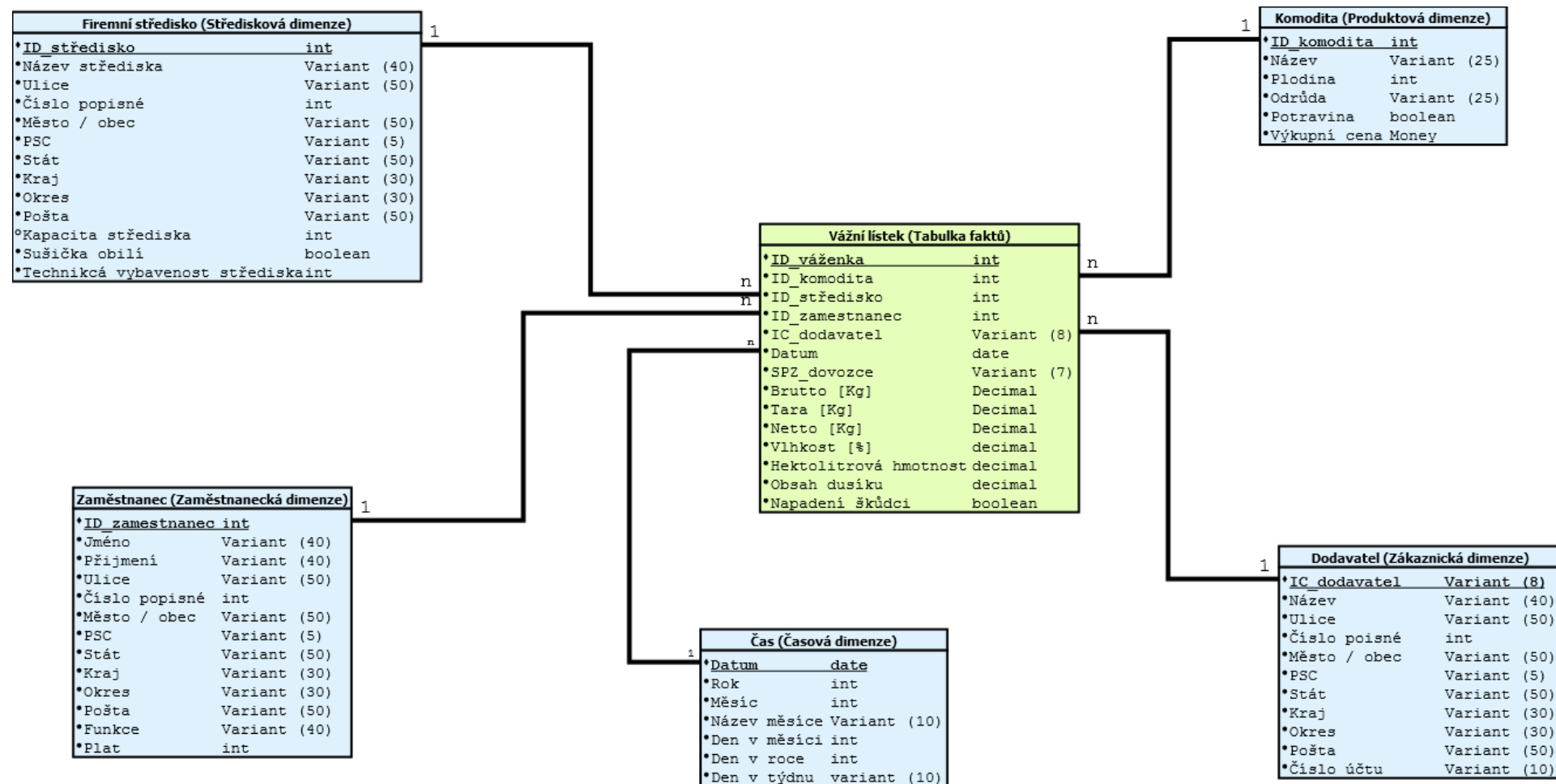
Relační model pro OLTP aplikaci

Ať už budeme data zadávat ručně, nebo použijeme nástroj pro naplnění, budeme potřebovat relační databázi pro OLTP aplikaci. Jak bylo popsáno v teoretickém úvodu, OLTP aplikace, resp. její databáze nám poskytne data pro zpracování multi-dimenzionální analýzy.



Obrázek 16 - Entito-relační diagram

Model hvězdy pro datový sklad



Obrázek 17 - Model Hvězdy

4. VLASTNÍ NÁVRHY ŘEŠENÍ

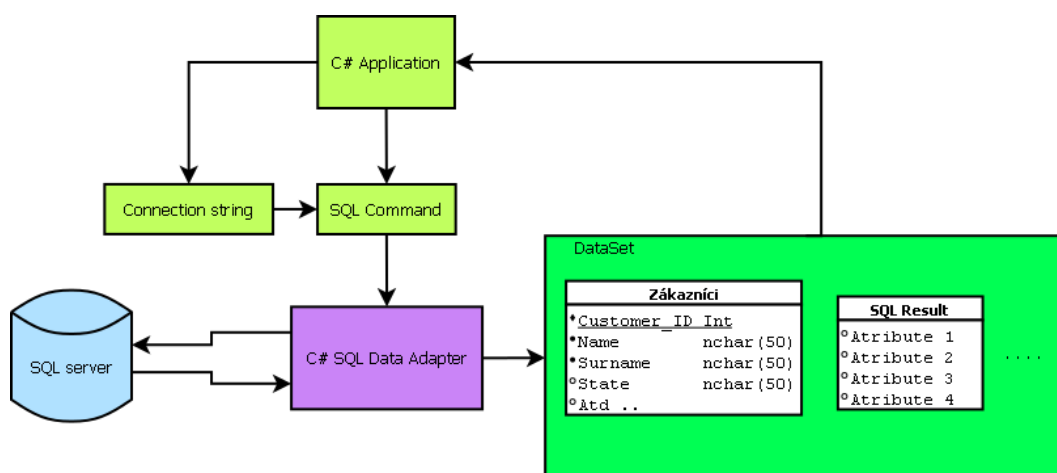
V této části budu popisovat jednotlivé moduly a fáze projektu.

4.1 OLTP Aplikace

Tato podkapitola obsahuje popis aplikace pro zachycování transakčních dat z hlavního procesu. Hlavní proces byl popsán v kapitole 2.3.

4.1.1 Datový adaptér (Navázání spojení s SQL serverem)

Dříve než vůbec začneme data ukládat nebo je naopak načítat z databáze, musíme se nejprve k této databázi připojit. Pro OLTP aplikaci slouží jako úložný prostor relační databáze realizovaná v prostředí MS SQL. Díky tomu, že programujeme celý projekt v jazyce C# pod platformou dot NET a MS SQL je také od firmy Microsoft, pak je připojení jednoduché a je realizováno pomocí tzv. Datového adaptéru.

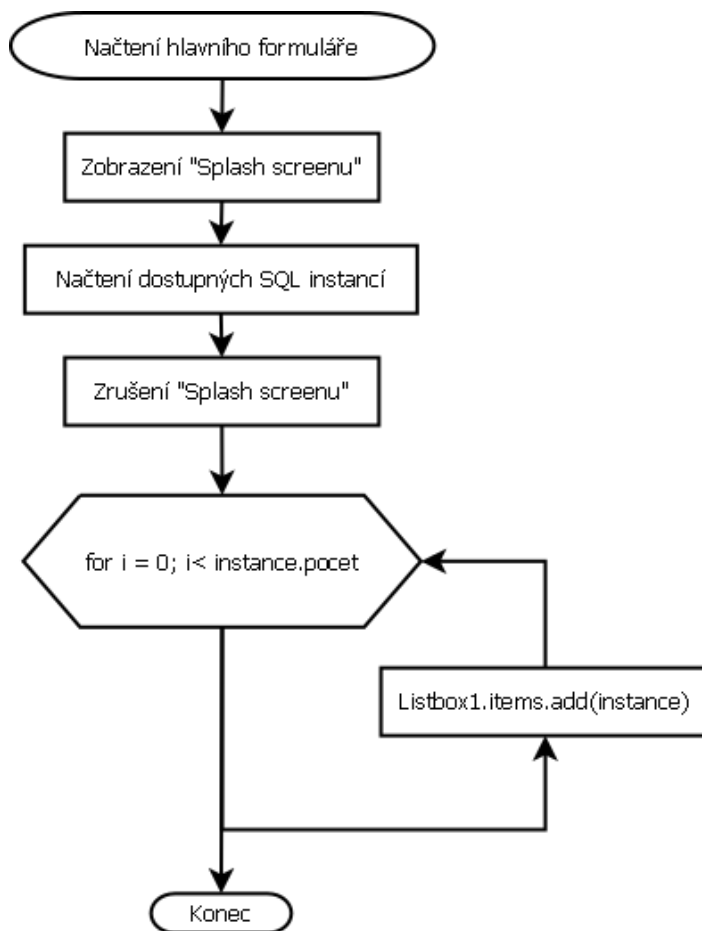


Obrázek 18 - Připojení k SQL databázi

Na obrázku je vidět datový adaptér jako fialový blok. Adaptér přijme od aplikace připojovací řetězec, ve kterém se dozví k jaké konkrétní databázi se připojit, dále mu aplikace předá příkaz v jazyce SQL a adaptér tyto údaje zpracuje a odešle na server. Od serveru pak přijme výsledky, které aplikaci zpátky prezentuje ve formě datasetu.

4.1.2 Výběr správné databáze

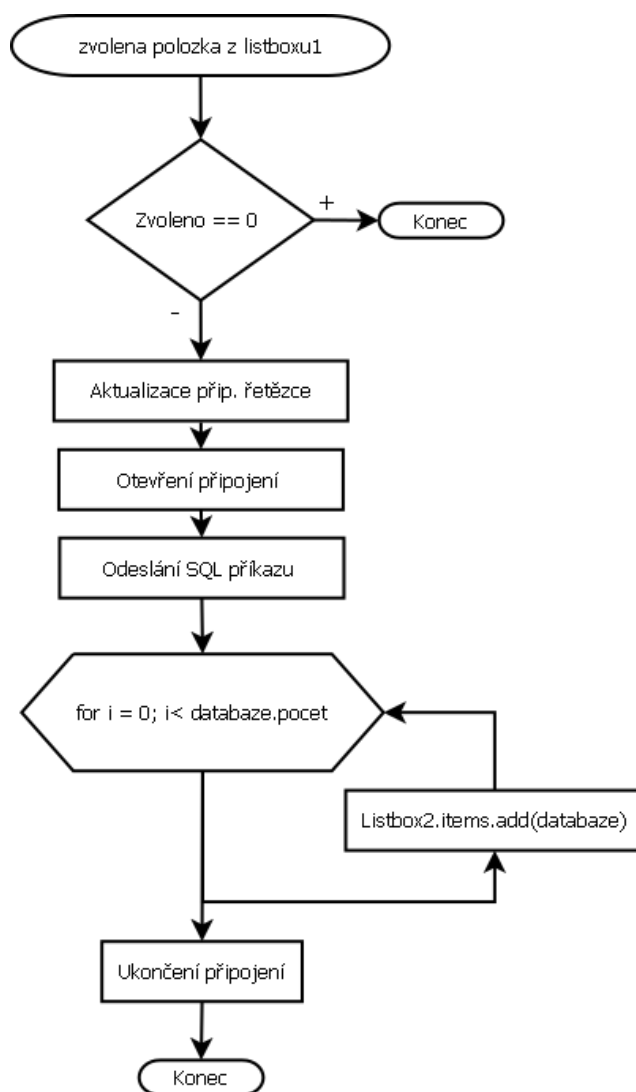
Je zcela běžné, že v rámci jedné běžící instance MS SQL Serveru, můžeme najít hned několik různých databází. Zároveň i během vývoje této aplikace, kdy se mění počítače na kterých je program testován, je vhodné mít možnost definování, ke které databázi se chceme připojit. Navíc se tím vyhneme možnosti připojení k nesprávné databázi.



Obrázek 19 - Diagram načítání SQL instancí

První řádky kódu se tedy spouštějí hned při načítání hlavního formuláře, tedy událost `form_load`. Protože vyhledání běžících SQL instancí zabere řádově několik vteřin, během kterých počítač, resp. aplikace vůbec nereaguje, rozhodl jsem se zobrazit tzv. „splash screen“, který upozorní na probíhající proces hledání. Po dokončení je opět zrušen a do listboxu se pomocí FOR cyklu načítají nalezené instance.

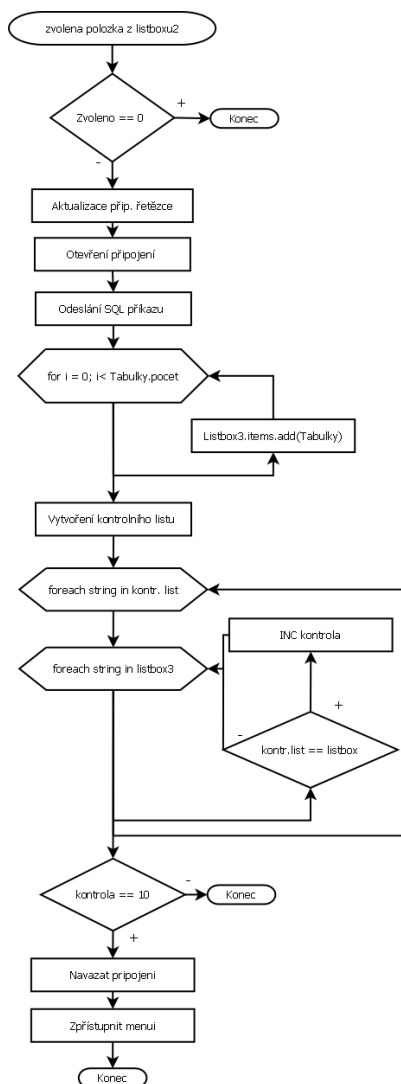
Poté co jsme načetli aktivní instance a vypsali je do listboxu č. 1, budeme čekat, až uživatel vybere některou z nich. V momentě, kdy uživatel zvolí jednu z instancí, spustí se událost „listbox1_SelectedIndexChanged“.



Obrázek 20 - Diagram načtení databází

Hned v úvodu procesu, kontrolujeme, zda není vybrána neplatná hodnota v listboxu. Poté se vytváří nový připojovací řetězec na základě předchozí vybrané instance, tedy hodnota listboxu č. 1. Poté je otevřeno spojení a vyslán SQL příkaz na výpis databází v rámci zvolené instance. Poté jsou opět pomocí cyklu FOR načítány jednotlivé databáze do dalšího listboxu.

V tuto chvíli tedy víme, k jaké se chceme připojit instanci SQL serveru a máme v listboxu č. 2 vypsané všechny dostupné databáze. Posledním krokem bude vybrat konkrétní databázi. Budeme tedy čekat, až uživatel vybere některou z databází vypsanou v listboxu č. 2. V momentě, kdy vybere, spustí se poslední událost „listbox2_SelectedIndexChanged“.



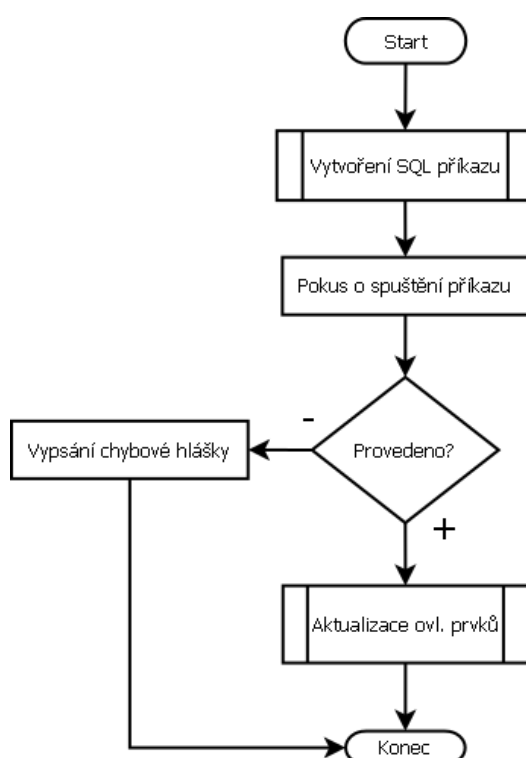
Obrázek 21 - Kontrola vybrané databáze

Zde se opět zkontroluje, zda není vybrána položka s neplatným indexem a poté se vytvoří aktualizovaný připojovací řetězec, tentokrát s konkrétní instancí i databází. Opět se vyšle SQL příkaz, který protentokrát získá tabulky obsažené v konkrétní databázi. Doposud je proces velmi podobný tomu předcházejícímu, avšak v tomto případě potřebujeme zkontrolovat, zda tabulky obsažené v dané databázi odpovídají

požadavkům našeho programu. V další části je tedy vytvořen kontrolní seznam tabulek, které musí databáze obsahovat, aby bylo možné se k ní připojit. Pak už jenom zbývá prohledat obsah databáze, zda obsahuje všechny požadované tabulky, v našem případě jsem zvolil cyklus FOREACH s vnořeným druhým cyklem FOREACH. Pokaždé, když je nalezena shoda, je kontrolní čítač navýšen. Pokud v závěru čítač obsahuje správný počet, je navázáno trvalé spojení a zpřístupněno hlavní menu programu. Ovšem, může také nastat situace, kdy není možné nalézt instanci SQL serveru, může to být v takové situaci, kdy je nainstalována verze „Express“, nebo pokud nejsou spuštěny některé služby SQL serveru. V takovém případě je možné v hlavním menu zadat ručně připojovací řetězec.

4.1.3 Vytváření nových záznamů

Úkolem OLTP aplikace je ukládání transakčních dat, tedy vytváření nových údajů v databázi bude nejčastějším úkolem tohoto modulu.



Obrázek 22 - Ukládání záznamů

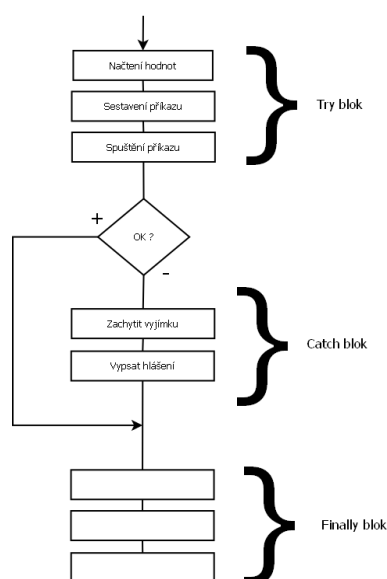
Pro úsporu místa bylo vytvoření SQL příkazu vyobrazeno jako sub-proces. SQL příkaz je vytvářen tak, že se postupně sbírají potřebné řetězce a hodnoty z ovládacích prvků formuláře kombinují se s připraveným SQL příkazem. Poté je proveden pokus o spuštění příkazu pomocí bloku TRY. V případě, že je příkaz proveden, je spuštěn další sub-proces, a to aktualizace hodnot v ovládacích prvcích. Pokud ovšem nastane problém, vypíše se chybové hlášení a proces je ukončen.

4.1.4 Kontrola správnosti dat

Kvalita dat, jejich správnost a smysluplnost má v této problematice zcela zásadní význam. Pokud bychom měli databáze a datové sklady naplněny nekvalitními daty, ani analýzy by neposkytovali kvalitní informace. V této kapitole budou popsány metody a postupy pro kontrolu kvality ukládaných záznamů.

Try catch blok

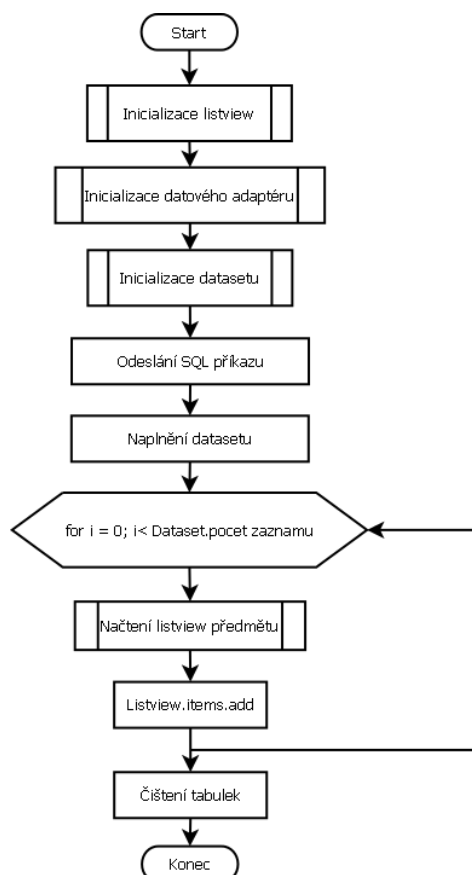
„Try catch“ blok v programování slouží ke spouštění kódu, při kterém může dojít k chybě, resp. neočekávané situaci. V části „try“ je vložen libovolný kód, v případě že je spuštěn bez problémů pokračuje program dále, v opačném případě jsou provedeny příkazy v části „catch“. Část „finally“ je spouštěna bez ohledu na to, zda byly příkazy v bloku „try“ provedeny úspěšně.



Obrázek 23 - Try blok

Jak nám ale tato metodika pomůže odhalit špatně zadaná data, když špatná hodnota v textboxu není ve své podstatě chyba? Jelikož pracujeme s SQL serverem, museli jsme si již na počátku definovat strukturu databáze, její entito-relační model, vazby a datové typy jednotlivých sloupců. Čili SQL server má definované datové typy a v případě, že dostane k uložení neplatnou hodnotu, ukončí příkaz a vypíše chybu. Jelikož nepracujeme přímo na SQL serveru, ale příkazy odesílá C# aplikace, musíme kontrolovat, zda jsou hodnoty v souladu s datovým modelem. Pokud část kódu, která obsahuje povely k odeslání a vykonání SQL příkazu umístíme do bloku „try catch“, pak v případě špatných dat SQL server vrátí chybové hlášení a aplikace zachytí toto hlášení v části „catch“. Chyba je tedy odhalena až v momentě, když vytvořený příkaz zpracovává SQL server, tím pádem nemusíme data testovat z různých hledisek (datový typ, délka, prázdný záznam apod.). Navíc je v zachyceném chybovém hlášení uveden i důvod odmítnutí hodnoty.

4.1.5 Zobrazování uložených záznamů

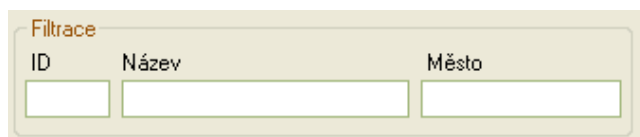


Obrázek 24 - Zobrazování záznamů

Při práci s OLTP aplikací potřebujeme průběžně zobrazovat uložená data. Tento proces je částečně popsán v úvodu této kapitoly (Datový adaptér). Pomocí datového adaptéru je tedy odeslán příkaz SQL a přijat výsledek do „datasetu“. Podle počtu řádků v tabulce datasetu je vytvořen cyklus, který postupně načítá záznamy do ovládacího prvku listview. V závěru po načtení je obsah tabulky vymazán.

4.1.6 Filtrace uložených záznamů

Jelikož může být databáze partnerů nebo i jiných entit velice rozsáhlá, v našem případě je to přes tisíc záznamů, může být práce nepřehledná. Proto je filtrace nezbytnou součástí uživatelského rozhraní OLTP aplikace. Z celkové množiny záznamů je pak možné vybírat podle určitých kritérií. Principiálně se jedná o naprosto stejný proces jako v předchozím případě, tedy načítání dat pomocí datového adaptéru, avšak v tomto případě se SQL serveru předává jiný příkaz. Příkaz se liší v klauzuli „Where“, kde je navíc operátor „Like“, který upraví vyhledávání tak, že se vyhledávají záznamy obsahující i část zadaného výrazu. Jelikož by byl diagram prakticky stejný nebudu jej zde uvádět.

The image shows a web form with a light beige background and a thin green border. At the top left, the word "Filtrace" is written in a small, orange-brown font. Below this, there are three input fields arranged horizontally. Each field has a label above it: "ID", "Název", and "Město". The "ID" field is a small, narrow box. The "Název" field is a medium-width box. The "Město" field is a small, narrow box. All three fields are currently empty and have a light gray border.

Obrázek 25 - Rozhraní pro filtraci

Filtrovat je možné i na základě více kritérií. Například jak je zobrazeno na obrázku 22, budou nás zajímat záznamy s určitým názvem a z určitého města. Potom jsou použity logické spojky „AND“, které zajistí restrikci podle více parametrů. V případě že pole není vyplněno, daný parametr není brán na zřetel. Co se týče programového řešení, celý sled příkazů je deklarován jako procedura, která je spouštěná při změně hodnoty některého z textboxů, tedy event „OnTextBox_text_changed“.

4.1.7 Editace uložených záznamů

Tento proces je opět velice podobný, již popsanému procesu, vytváření nového záznamu, kde se liší pouze SQL příkaz.

4.2 Plnění OLTP Aplikace

Aplikace OLTP je nyní funkční a schopna vytvářet nové záznamy. Abychom mohli provést analýzu, musíme naplnit OLTP aplikaci daty, resp. z OLTP aplikace poté naplníme datový sklad, ze kterého pomocí MS Business intelligence studia provedeme multi-dimenzionální analýzu.

4.2.1 Získání potřebných dat

Jak je zmíněno v kapitole charakteristika společnosti, společnost ZZN Pelhřimov a.s. souhlasila s poskytnutím transakčních dat z útvarů pro skladování zemědělských komodit s podmínkou, že data budou maskována tak, aby zveřejnění žádným způsobem neohrozilo společnost. Data o příjmu komodit byla vyexportována ve formě XLS souboru (MS Excel), který obsahuje cca 65 000 záznamů. Takto vyexportovaná data obsahují pouze cizí klíče na ostatní entity (Partneři, sklady, komodity a zaměstnanci). Abychom tedy mohli provádět analýzy, bylo nutné získat i databáze právě uvedených entit. Z informačního systému ACC bylo možné tyto informace získat ve formátu HTML, který bylo následně možné transformovat do XLS souboru, nicméně kvůli převádění nebyla zachována konzistentní struktura řádků a sloupců a bylo nutné data upravovat. Přesný popis problému a jeho řešení popíši v následující kapitole.

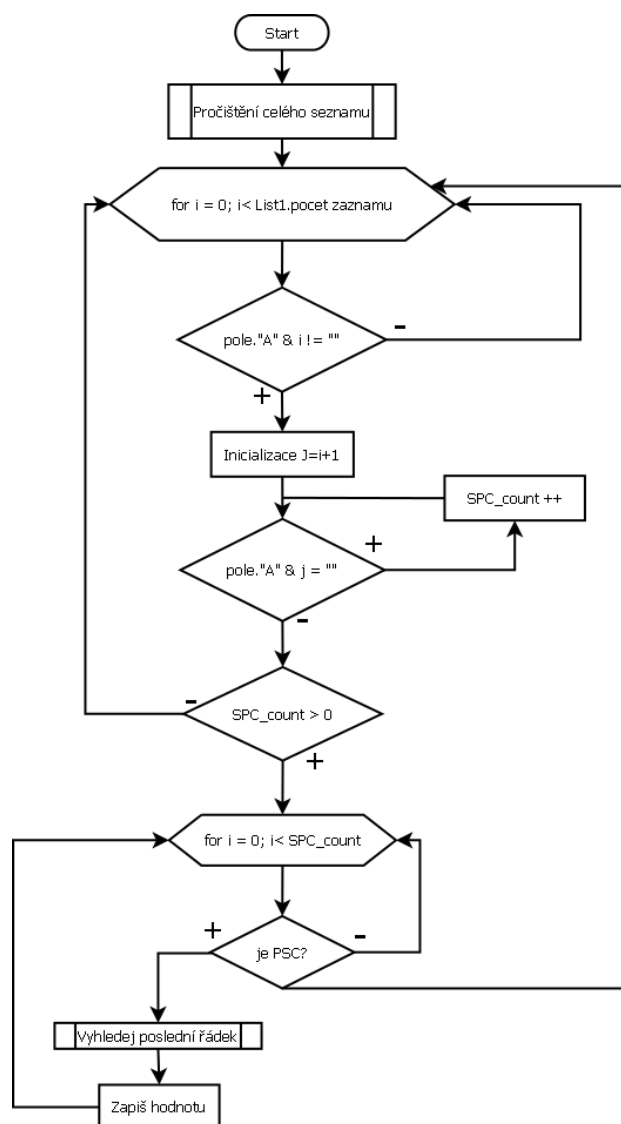
4.2.2 Korekce dat

Získaná data bylo nutné upravit, neboť neměli konzistentní strukturu, ze které by se nechali čerpat informace.

Kód partnera	Partner A	Partner A
		Ulice 1
		381 01 Cesky Krumlov
Kód partnera	Partner B	Partner B
		Ulice 2
		374 01 Trhove Sviný
Kód partnera	Partner C	Partner C
Kód partnera	Partner D	Partner D
Kód partnera	Partner E	Partner E
Kód partnera	Partner F	Partner F

Tabulka 4 - Struktura databáze partnerů

Jak je možné vidět v tabulce, mezi záznamy je nepravidelný počet prázdných řádků a u některých záznamů není údaj o adrese vůbec. Bylo tedy nutné vytvořit konzistentní seznam partnerů s jejich adresami, resp. PSČ. Jelikož byla tato databáze uložena v XLS souboru, využil jsem prostředí VBA integrované v MS Excelu.

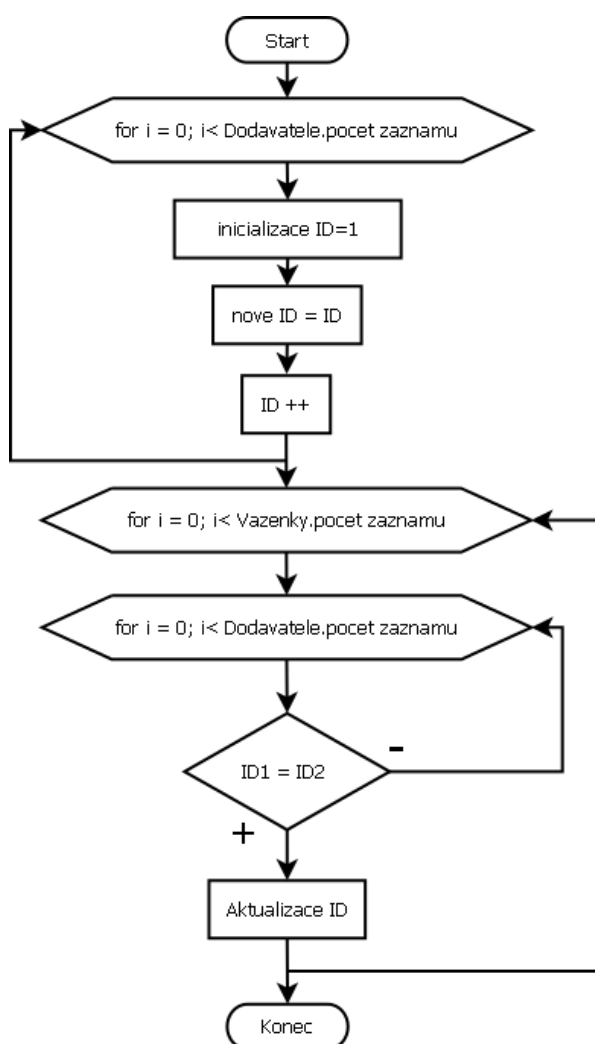


Obrázek 26 - Proces vyhledávání PSČ

Před zahájením celého procesu je nejdříve celý list očištěn o prázdné řádky, poté jsou vyhledávány neprázdné řádky. V momentě, kdy je nalezen neprázdný řádek, je spočítán počet prázdných řádků pod ním. Dále jsou zkontrolovány všechny řádky vedle prázdných řádků, zda neobsahují pětimístné číslo. Pokud je v buňce nalezen pětimístný číselný údaj, je zapsán na nový řádek nového seznamu společně s kódem partnera.

Stejným nebo podobným způsobem jsou zpracovány i ostatní entity (Zemědělské komodity, střediska firmy, zaměstnanci.)

Další potřebnou úpravou byla změna primárního klíče. Databáze partnerů využívala jako primární klíč textový řetězec, který v některých případech obsahoval několik po sobě jdoucích mezer. Při manipulaci s daty pak hrozilo odstranění přebytečných mezer a následného narušení integrity dat a relačního provázání s databází váženek.



Obrázek 27 - Přiřazování nových ID

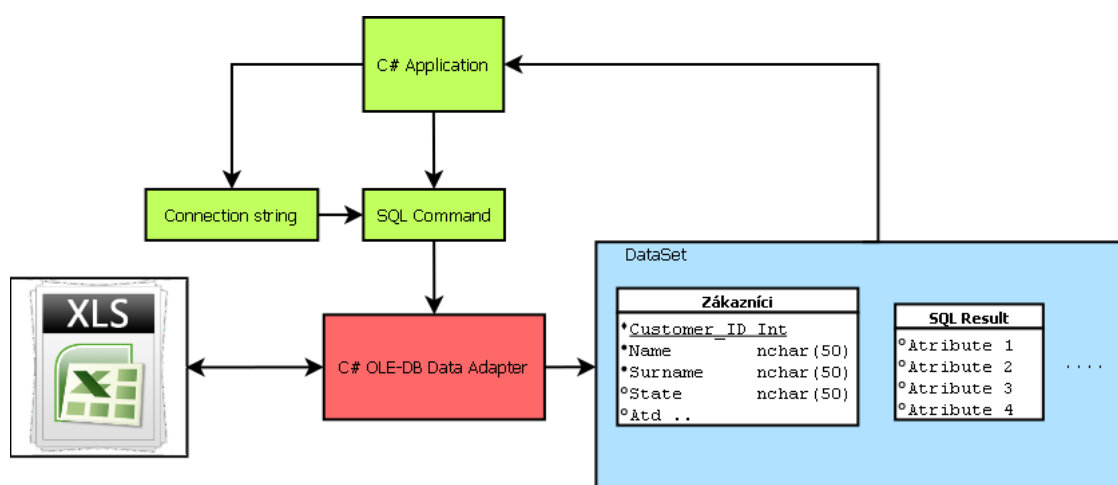
Nejprve je tedy proveden cyklus, který postupně prochází celý seznam partnerů a přiřazuje jim nové identifikační čísla. Poté další cyklus prochází postupně každý záznam váženky resp. sloupce s uvedeným partnerem, od kterého je daná dodávka a

porovnává jej s každým záznamem seznamu partnerů. Tedy cyklus s vnořeným cyklem, v momentě kdy je nalezena shoda, je přiřazené nové ID.

4.2.3 Načtení XLS souborů

V tuto chvíli máme tedy fungující OLTP aplikaci a ošetřená data pro naše potřeby. Zbývá tedy data načíst do relační databáze OLTP aplikace. Data by bylo možné importovat na SQL server přímo v SQL management studiu, nebo pomocí integračních balíčků v Business intelligence studiu. To by však znamenalo, že každý uživatel by musel ovládat jazyk SQL, nebo alespoň znát prostředí BI studia. Proto jsem se rozhodl pro import dat vytvořit modul, pomocí kterého bude možné importovat data z XLS souborů bez znalosti SQL nebo BI studia.

Připojení k XLS souboru.

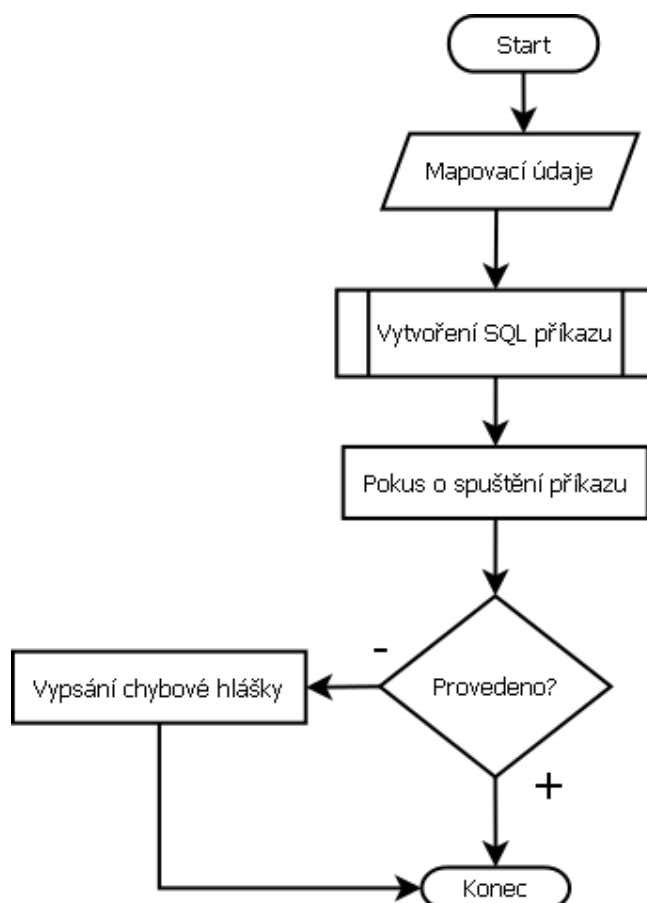


Obrázek 28 - Připojení k XLS souboru

Hned z tohoto obrázku je patrná velká podobnost se způsobem připojení k SQL databázi uvedeným v úvodu třetí kapitoly. Skutečně se principiálně jedná o naprosto stejný postup s tím rozdílem, že v připojovacím řetězci je cesta k souboru a datový adaptér je typu OLE-DB, ne SQL. Přestože se jedná o OLE-DB datový adaptér, využívá se zde SQL jazyka pro načítání dat, tedy příkaz SELECT. Výsledek je pak stejně jako v minulém případě načítán do tabulek v datasetu.

4.2.4 Uložení dat z XLS souborů

Přístup k datům uloženým v XLS souboru již máme a teď je potřebujeme přenést do naší relační databáze.



Obrázek 29 - Import záznamů z XLS

Opět je tento proces podobný s jedním již popsaných procesů, konkrétně ukládání dat, avšak s tím rozdílem, že zde musíme před spuštěním specifikovat, ve kterém sloupci jsou jaká data. Poté co je příkaz spuštěn, neaktualizují se žádné ovládací prvky, neboť zobrazení je prováděno na jiném formuláři, kde se o načtení postará jiný proces.

4.3 ETL Fáze

V tuto chvíli, máme již OLTP aplikaci naplněnu daty a můžeme se soustředit na transfer dat do datového skladu. Samotné nahrávání dat je až fáze „loading“, jak bylo popsáno v teoretickém úvodu a je jí věnovaná další kapitola.

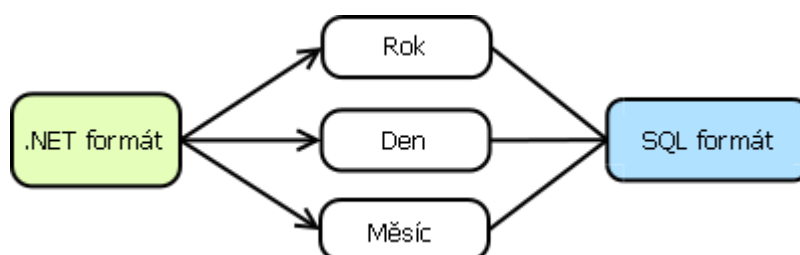
4.3.1 Extrakce

Fáze extrahování zde v praktické části není moc zřetelná, neboť jsme data obdrželi v jediném souboru. Mohlo by se tedy zdát, že jsme data extrahovali z jediného zdroje. Ve skutečnosti data pochází z více jak 17 firemních středisek.

4.3.2 Transformace

Za předpokladu, že všechna firemní střediska budou využívat stejnou OLTP aplikaci, nehrozí problém rozdílného formátu dat. Data jsou kontrolována nejprve při ukládání do relační databáze samotnou OLTP aplikací a podruhé při importu do OLTP aplikace. V případě, že by některé středisko využívalo jinou OLTP aplikaci, musel by se navrhnout speciální modul pro transformaci konkrétních dat.

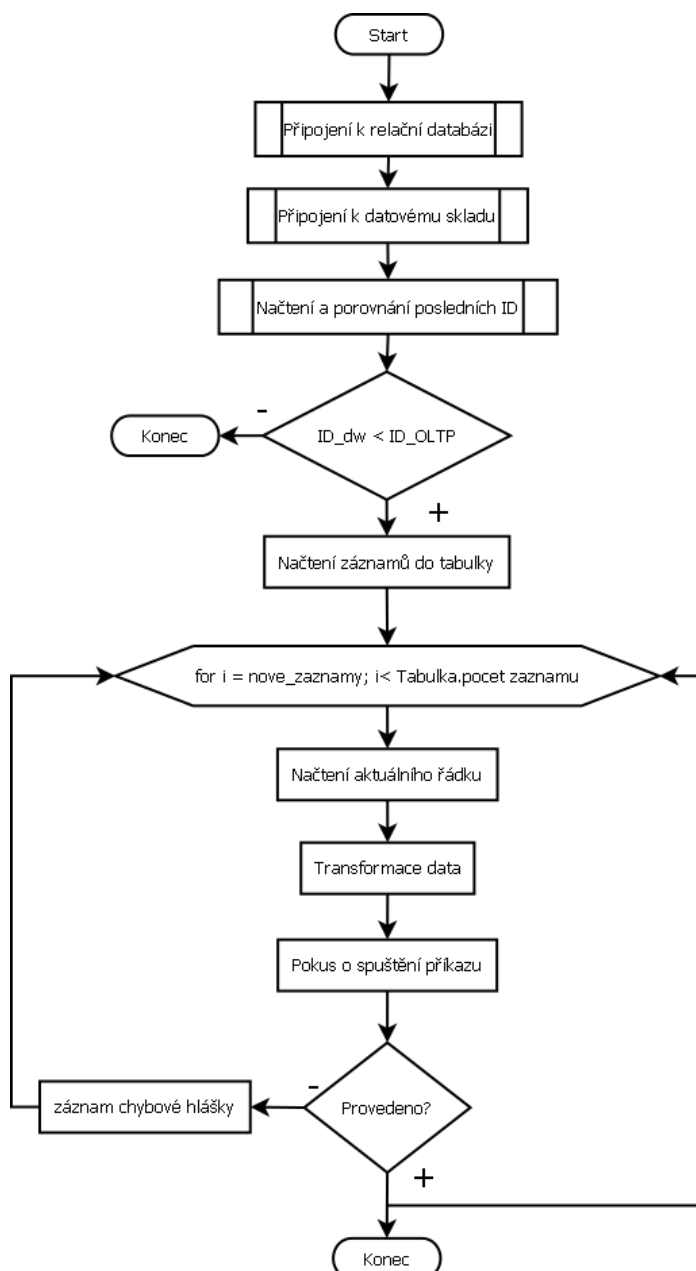
Nicméně přestože by všechna střediska používala stejnou OLTP aplikaci, může nastat situace, kdy se bude lišit formát kalendářního data (pořadí dne, měsíce a roku). Je to způsobeno tím, že formát kalendářního data, který vrací ovládací prvek „DateTimePicker“ se řídí podle geografického nastavení operačního systému. Tedy pravděpodobnost, že na některém z počítačů může být jiný formát data je vysoká a musí se s tímto problémem počítat.



Obrázek 30 - Konverze formátů času

Jak je uvedeno na obrázku 27, ačkoliv se formát může lišit i v OLTP aplikaci, pořád je toto datum v „dot NET“ formátu. To znamená, že je možné z toho data získat jednotlivé údaje odděleně (den, měsíc, rok). Poté je to pouze správné seřazení textových řetězců tak, abychom získali formát, který potřebujeme a odeslali jej na SQL server. Konverze pak mezi typem dot Net a SQL je otázkou jednoho příkazu.

4.3.3 Nahrávání (Loading)



Obrázek 31 - Nahrávání záznamů do datového skladu

Abychom mohli odesílat data do datového skladu, musíme se připojit k oběma databázím, OLTP i datovému skladu. Poté je provedena kontrola, zda záznamy vůbec potřebují aktualizovat, neporovnává se počet záznamů, ale výše posledního ID v obou databázích. Jestliže je ID v datovém skladu nižší nežli ID v OLTP databázi, může se zahájit transfer. Načtou se tedy potřebné záznamy a FOR cyklus se nastaví na správný počet opakování. Během každého cyklu je načtena hodnota, v případě potřeby je transformována a uložena do datového skladu.

4.3.4 Tabulky dimenzí

Proces popsaný v předchozí podkapitole má na starosti plnění právě tabulek dimenzí i faktů. Přesné schéma jsme si již uvedli. Až na časovou dimenzi, všechny tabulky vycházejí z tabulek v relační databázi OLTP aplikace, pouze s tím rozdílem, že je zde redundance. Tabulka časové dimenze je pak vytvářena na základě unikátních kalendářních dat. Dalším rozdílem oproti databázi OLTP je vytváření primárních klíčů, v OLTP aplikaci jsou automaticky generovány primární klíče inkrementací čísla INT. Zde je primární klíč pouze přejat, jelikož v opačném případě by nebyla zachována integrita dat. Na závěr je dobré zmínit, že nejprve jsou vytvářeny tabulky dimenzí, neboť tabulka faktů se odkazuje pomocí cizích klíčů právě na tabulky dimenzí. Nemůže se tedy odkazovat na neexistující záznamy, a proto jsou tabulky dimenzí vytvářeny jako první.

4.3.5 Tabulka faktů

Tabulka faktů je prakticky nezměnná tabulka váženek z OLTP aplikace. Stejně jako u tabulek dimenzí, ID se zde pouze přejímá od OLTP aplikace. Sloupce, které se mohou využívat pro analýzy, jsou: Brutto, tara, netto, vlhkost, obsah dusíku a hektolitrová hmotnost.

4.4 Analýzy

Po úspěšném exportu záznamů do datového skladu můžeme začít se sestavováním datové kostky a poté s prováděním analýz. Jak bylo již zmíněno, k sestavení datové kostky i analýzám bude využito Business intelligence studio do firmy Microsoft.

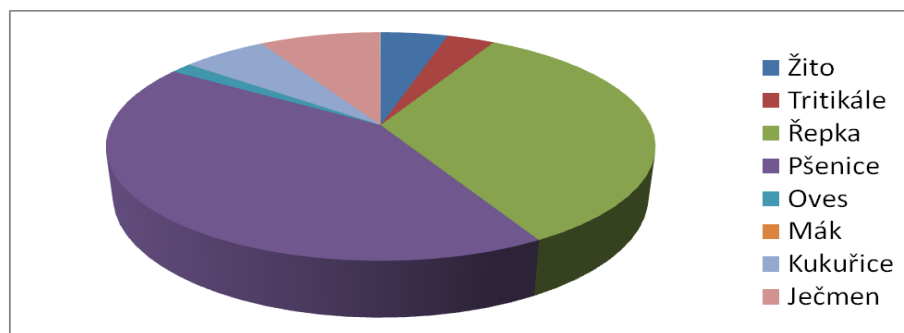
4.4.1 Analýza geografického rozložení komodit

- Využité dimenze: Komodita – plodina; Dodavatel – Kraj; Čas – roky.
- Využitá fakta: Váženka – Netto.
- Typ datové kostky: 3D.
- Přínosy: Na základě této analýzy můžeme predikovat zájem o hnojiva a osiva v nadcházejících obdobích a patřičně tomu přizpůsobit zásobování a marketing.

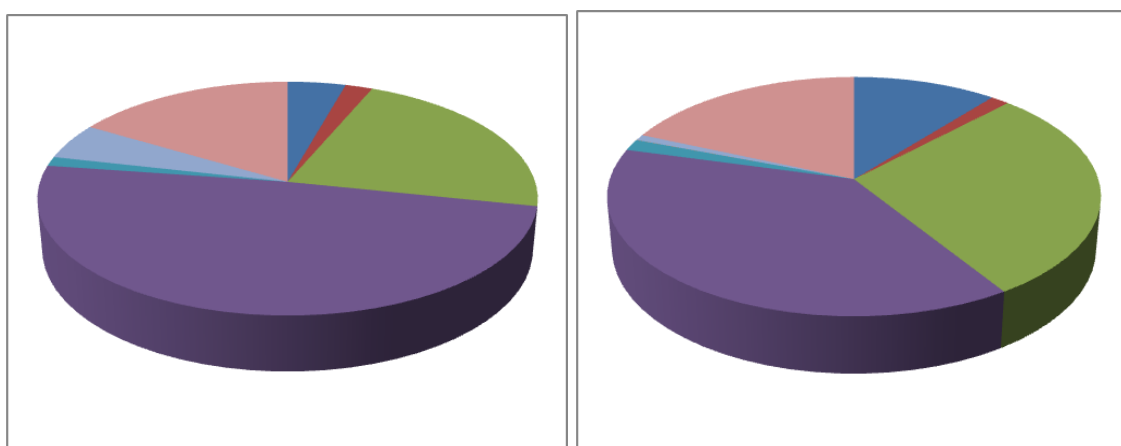
	Kraj Rok 2011									Grand Total
	Jihočeský	Jihomoravský	Olomoucký	Pardubický	Plzeňský	Praha	Středočeský	Vysočina	Zlínský	
Plodina	Netto	Netto	Netto	Netto	Netto	Netto	Netto	Netto	Netto	Netto
Žito	4969,6					2862,7	2602,7	6351,2		16786,2
Tritikále	3540,2		28,7				1220,0	826,8		5615,6
Řepka	35435,6	44,7	7,6		243,1	723,4	13278,0	16818,5		66551,0
Pšenice	44235,3	193,8	5,3		141,1	2448,5	29575,6	22409,1	639,7	99648,4
Oves	1560,9				16,4		777,7	901,3		3256,2
Mák	19,2							10,1		29,3
Kukuřice	6247,8	1209,1		1042,1		110,8	3026,5	455,3		12091,6
Ječmen	8957,2		27,2		94,3	154,5	9906,7	10839,3		29979,3
Hrách	55,4	32,4								87,8
Grand Total	105021,4	1479,9	68,9	1042,1	495,0	6299,9	60387,2	58611,5	639,7	234045,5

Tabulka 5 - Výstup MS BIDS studia

V BI studiu byla vytvořena datová kostka z uvedených dimenzí a faktů. BI studio poté zobrazuje výsledky ve formě výše uvedené kontingenční tabulky, aby mělo takovéto analýzy smysl provádět, je důležitá jejich správná prezentace. V další části zobrazíme výsledky pomocí grafů. Pro další analýzy budu v rámci úspory místa vkládat pouze grafické výsledky analýz.



Obrázek 32 - Výsledky pro Jihočeský kraj



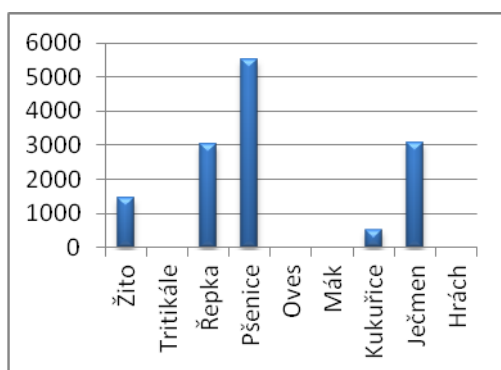
Obrázek 33 - Výsledky Středočeského kraje a kraje Vysočina

Jak lze vidět na grafech, v největší míře je ve všech krajích zastoupena pšenice, aby mohli zemědělci v příštím roce opět úspěšně pěstovat, musejí správně půdu pohnojit a zasít nové plodiny, tedy tato analýza nám poskytne informace o jaká hnojiva a osiva bude zájem a zároveň tyto informace můžeme použít při poradenství.

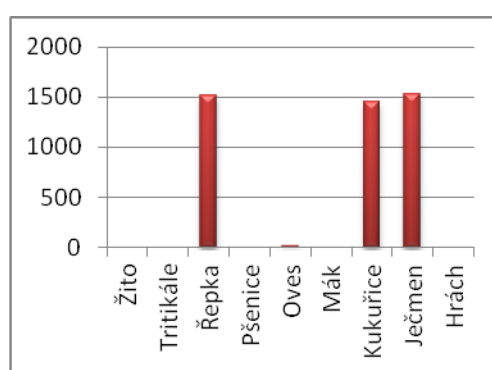
4.4.2 Analýza dodavatelů

- Využité dimenze: Komodita – plodina; Dodavatel – ID; Čas – roky.
- Využitá fakta: Netto [t].
- Typ datové kostky: 3D.
- Přínosy: Tato analýza nám poskytne informace o celkové velikosti dodávek komodit za minulý rok od jednotlivých dodavatelů. Na základě toho můžeme plánovat budoucí transakce s dodavateli.

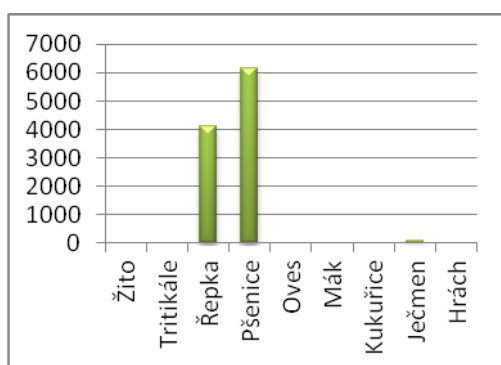
ID 0353



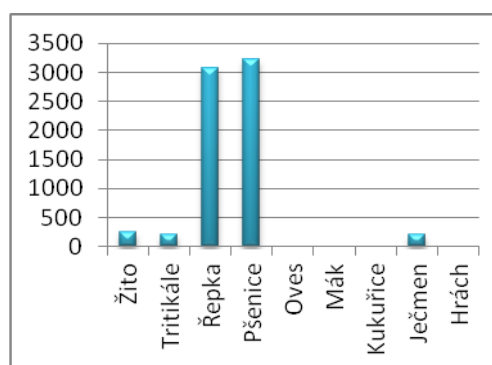
ID 0477



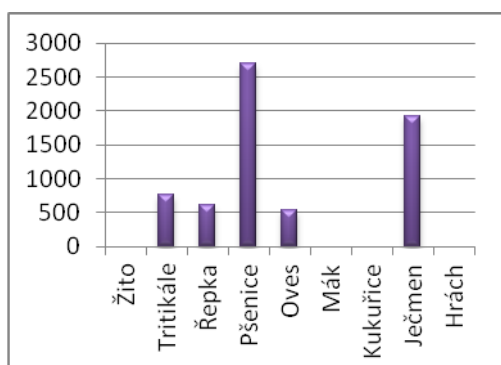
ID 0595



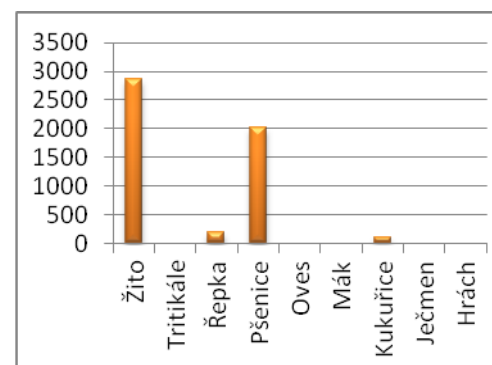
ID 0513



ID 1070



ID 0758



Tato analýza nám může posloužit při plánování výroby a poskytnout nám podklady, který dodavatel je vhodný na dodání určité komodity.

4.4.3 Analýza firemních středisek

- Využité dimenze: Středisko – ID středisko, Komodita – plodina, Čas – roky.
- Využitá fakta: Netto.
- Typ datové kostky: 3D

- Přínosy: Takto můžeme analyzovat, jaké firemní střediska mají zkušenosti se skladováním určitých komodit.

ID Střediska	Specializace na komoditu
2	Tritikále
2	Pšenice
2	Mák
3	Ječmen
5	Žito
7	Ječmen
14	Řepka
14	Hrách
15	Oves
15	Kukuřice

Střediska, která jsou uvedena v tabulce resp. pracovníci, kteří v nich pracují mají největší zkušenosti s manipulací a správným skladováním zmíněných komodit. Takové informace jsou velice cenné, a pokud se podaří je získat do znalostní báze podniku, přispěje to k efektivitě všech skladů.

4.4.4 Analýza velikosti dodávek

- Využité dimenze: Dodavatelé – ID; Čas – Roky; Střediska – ID
- Využitá fakta: Netto < 10.
- Typ datové kostky: 3D
- Přínosy: Na základě údajů z této analýzy můžeme oslovit partnery, kteří realizují velké množství dodávek s malou hmotností.

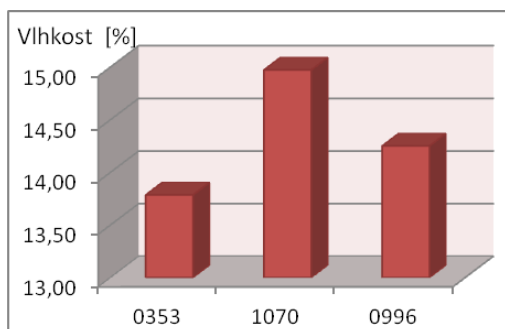
ID Dodavatele	Průměrná váha dodávky [t]	Počet dodávek v minulém roce
0353	6,444	161
1070	6,522	157
0494	6,532	148
0884	6,507	98
0514	6,378	82
0202	6,49	77

Tabulka 6 - Nejvhodnější partneři na nabídku dopravy

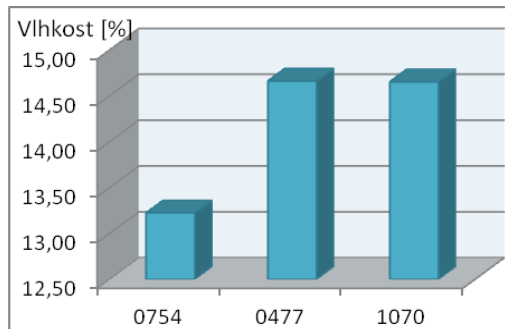
4.4.5 Analýza kvality dodávaných komodit

- Využité dimenze: Dodavatel – ID dodavatel; Komodita – plodina, Čas – roky.
- Využitá fakta: Netto, Vlhkost, Počet váženek (COUNT)
- Typ datové kostky: 3D.
- Přínosy: Zde budeme analyzovat největší dodavatele z hlediska vlhkosti dodávaných komodit.

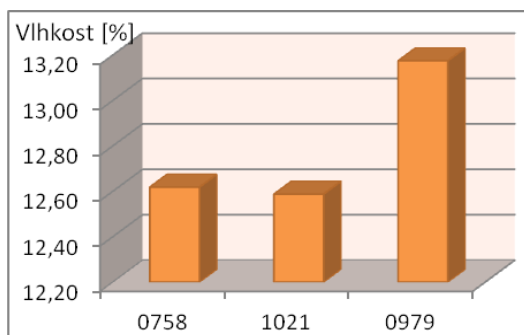
Přední dodavatelé ječmene



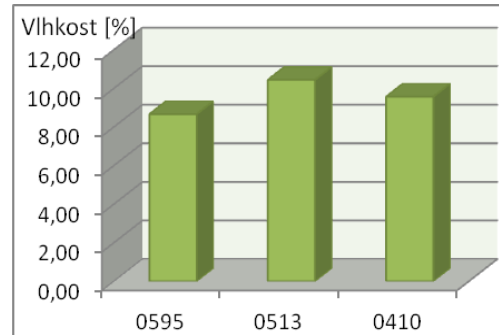
Přední dodavatelé pšenice



Přední dodavatelé žita



Přední dodavatelé řepky

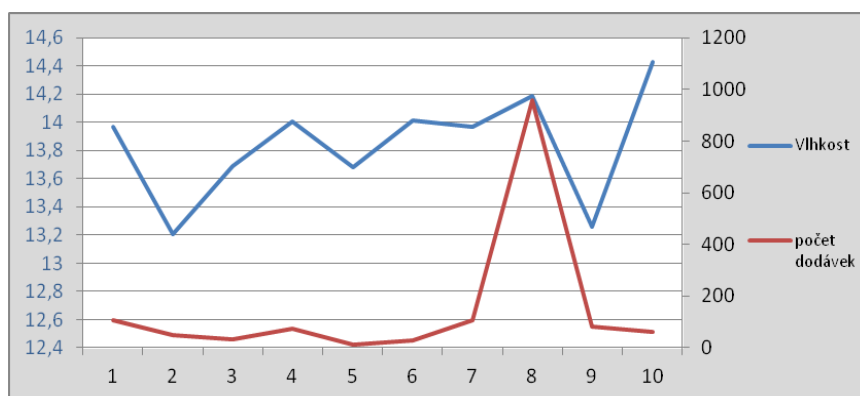


Čím nižší je vlhkost dodávané komodity, tím lepší jsou ostatní parametry jako obsah dusíku, klíčivost apod. Nízká vlhkost jednoznačně negarantuje dobré parametry, ale naopak vysoká vlhkost, značí téměř jistou pravděpodobnost, že ostatní parametry nebudou dosahovat standardů. Zároveň komodita s nízkou vlhkostí nepotřebuje dodatečnou péči a tím se snižují náklady. Z uvedených partnerů můžeme pak vybírat ty, kteří dodávají komodity s nejnižší vlhkostí. Naopak partnerům, kteří dodávají vysokou vlhkost, můžeme nabídnout poradenské nebo zpracovatelské služby.

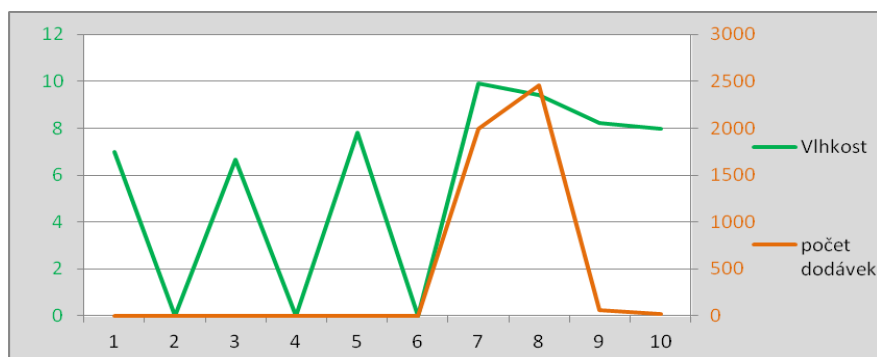
4.4.6 Analýza vlhkosti v čase

- Využité dimenze: Komodita – plodina; Čas – Roky/měsíce;
- Využitá fakta: Vlhkost, počet váženek.
- Typ datové kostky: 2D.
- Přínosy: Možné zjištění vhodné doby pro sklizeň resp. dodávku komodit a plánování příprav pracovišť.

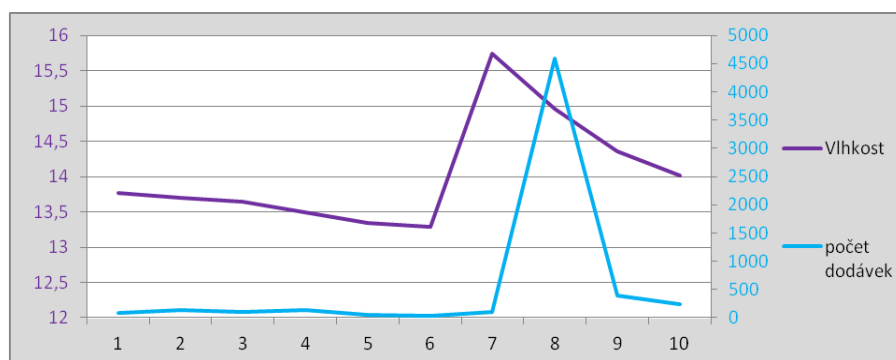
Analýza pro ječmen



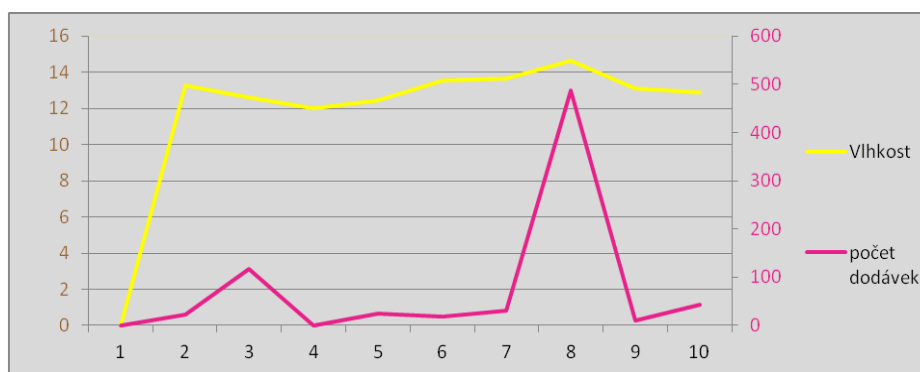
Analýza pro řepku



Analýza pro pšenici



Analýza pro žito



Podle grafů je možné stanovit čas, kdy je jaká komodita sklízena, resp. dodávána a jaká je její průměrná vlhkost. Částečně se tyto informace nechají využít při poradenství a také je možné díky těmto znalostem lépe plánovat najímání pomocné pracovní síly nebo plánování využití skladů.

4.5 Doporučení

4.5.1 Dodávka hnojiv, chemických přípravků a osiv

Analýzy ukázaly převažující zastoupení pšenice v pěstěných komoditách, to znamená zvýšený zájem o hnojiva, která se používají na hnojení právě po této komoditě. Mezi největší producenty pšenice patří dodavatelé:

ID 0353 – Okres Benešov; Středočeský kraj

ID 0595 – Okres Jihlava; Vysočina

ID 0513 – Okres České Budějovice; Jihočeský kraj

ID 1070 - Okres Benešov; Středočeský kraj

Tedy těmto partnerům vytvořit nabídku na dodání hnojiv. Zároveň jejich geografické rozložení nám poskytne podklady pro efektivní zásobování a distribuci pro sklady chemických hnojiv. Rovněž tak nabídka osiva, vhodného pro aplikaci po pšenici má velkou šanci na úspěch.

4.5.2 Uzavření obchodních dohod a smluv

Na základě plánu výroby oslovit některé dodavatele, kteří v loňských letech dodávali největší množství komodit. Jsou to:

ID 0353 – Pšenice, Řepka, Ječmen

ID 0477 – Řepka, Kukuřice, Ječmen

ID 0595 – Řepka, Pšenice

ID 0513 – Řepka, Pšenice

ID 1070 – Pšenice, Ječmen

ID 0758 – Žito, Pšenice

4.5.3 Podnikové kurzy / obohacení podnikové znalostní báze

Jednotlivá střediska, resp. její zaměstnanci mají zkušenosti s manipulací a skladováním různých komodit. Cílem je tedy vhodným způsobem přimět zaměstnance k sdílení těchto informací.

ID střediska	Nejlepší zkušenosti
2	Tritikále, Pšenice, Mák
3	Ječmen
5	Žito
7	Ječmen
14	Řepka, Hrách
15	Oves, Kukuřice

Absence ostatních středisek neznamena žádným způsobem sníženou kvalifikaci, nicméně uvedená střediska se specializují na uvedené komodity, tím pádem zaměstnanci mají lepší zkušenosti.

4.5.4 Nabídka služeb (Přeprava)

Uvedení partneři realizují velké množství dodávek s nízkou hmotností. Poskytnutím přepravy velkými nákladními vozy získají výhodu obě strany. Potenciální zákazníci jsou: ID 0353, ID 1070, ID 0494, ID 0884, ID 0514, ID 0202. Partneři budou tak schopni snížit náklady na přepravu komodit.

4.5.5 Příprava pracovišť, technologie a pomocné síly

Analýza dodávky komodit a jejich vlhkosti v čase nám odhalila časový úsek, kdy jsou dodávky nejfrekventovanější. Zároveň se ukázalo, že vlhkosti bývají vyšší spíše po začátku sklizňové špičky.

Z toho lze efektivně plánovat údržbu skladovací techniky, průmyslových sušiček a pomocné síly s ohledem na specializaci daného skladu. Zároveň je důležitá objednávka energie a pohonných hmot pro technologii skladovacích sil.

- Sklizňová špička ječmene vrcholí v osmém měsíci roku, vlhkost od začátku devátého měsíce klesá.
- Dodávky řepky jsou frekventované již v sedmém měsíci, vlhkost pak postupně klesá
- Pšenici dodávají dodavatelé nejvíce v osmém měsíci, vlhkost poté prudce klesá.
- Žito je dodáváno opět v osmém měsíci, ale vlhkost zde neklesá.

5. ZÁVĚR

Cílem této práce bylo vytvoření databázové aplikace a řešení pro business intelligence. Databáze pro ukládání transakčních dat byla realizována pomocí MS SQL serveru, nad kterou pracuje vytvořená aplikace v jazyce C#. V této aplikaci je možné nová data vytvářet nebo je importovat z existujících XLS souborů. Po vytvoření nebo importu transakčních dat je možné spustit plnění datového skladu. Poskytnutých 65 000 záznamů bylo zredukováno na přibližně 62 000 vlivem čištění a korekce dat. Po provedení analýz byly navrženy opatření ke zlepšení situace. Na základě geografického rozložení pěstěných komodit, byl doporučen adekvátní nákup hnojiv a osiv, dále bylo doporučeno s největšími dodavateli uzavřít obchodní smlouvy, analyzovány byly nejvhodnější střediska, resp. zaměstnanci, kteří mohou obohatit znalostní bázi podniku, pracoviště je možné na základě časových analýz včas připravit a nakonec byla doporučena nabídka přepravy pro skupinu partnerů, kteří realizují velké množství dodávek s malou hmotností. Dopad doporučených návrhů je obtížné kvantifikovat v peněžních jednotkách, nicméně navržené postupy rozhodně přispějí k hladkému chodu firemních procesů a zlepšení znalostní báze podniku.

6. SEZNAM ZDROJŮ

Knihy

- (1) HERNANDEZ, Michael; VIESCAS, John. *Myslíme v jazyce SQL*. 1. vydání. 2004. 380 s. ISBN 80-247-0899-X.
- (2) KOSEK, Jiří. *XML pro každého*. 1. vydání. 2000., 164s. ISBN 80-7169-860-1
- (3) CAO, Longbing; YU, Philip; ZHANG, Cheng; ZHANG, Huaifeng. *Data mining for Business Applications*. 1 vydání 2009. 302 s. ISBN: 978-0-387-79419-8
- (4) NOVOTNÝ, Ota; POUR, Jan; SLÁNKSY, David. *Business Intelligence, Jak využít bohatství ve vašich datech*. 1. vydání. 2004. 256 s. ISBN 80-247-1094-3
- (5) PARR, Olivia. *Data mining - Praktický průvodce dolováním dat*. 1. vydání. 2002. 330 s. ISBN 80-7226-577-6.
- (6) RAINARDI, Vincent. *Building a Data Warehouse with examples in SQL server*. 1 vydání 2008. 523 s. ISBN 1-59059-931-4
- (7) SHARP, John. *Microsoft visual C#*. 1. vydání. 2010. 696 s. ISBN 978-80-251-3147-3.
- (8) SILVERS, Fon. *Building and Maintaining a Data Warehouse*. 1. vydání 2008. 303 s. ISBN 1-4200-6462-2
- (9) VERCELLIS, Carlo. *Business Intelligence: Data mining and optimization for decision making*. 1. vydání 2009. 420 s. ISBN 978-0-470-51138-1

Internetové portály

- (10) ZZN Pelhřimov a.s. [online]. 2012 [Cit. 2012-03-25] Dostupné na WWW: <<http://www.zznpe.cz>>
- (11) MSDN Library [online]. 2012. [Cit. 2012-03-25] Dostupné na WWW: <<http://msdn.microsoft.com/en-us/library/>>

Materiály z absolvovaných kurzů

- (12) Ing. DYDOWICZ Petr, Ph.D., *Moderní programovací techniky* – Prezentace z přednášek zimní semestr 2010/2011, FP VUT).
- (13) Ing. KŘÍŽ Jiří, Ph.D., *Business intelligence* – Prezentace z přednášek letní semestr 2011/2012, FP VUT).

Seznam obrázků

Obrázek 1 - Možnosti kompilace	16
Obrázek 2 - Moduly visual studia	17
Obrázek 3 - Architektura BI	20
Obrázek 4 - Architektura datového skladu	25
Obrázek 5 - Schéma hvězdy	27
Obrázek 6 - Schéma sněhové vločky	28
Obrázek 7 - Schéma souhvězdí	28
Obrázek 8 – Metoda velkého třesku	29
Obrázek 9 – Metoda přírůstková	30
Obrázek 10 - ETL proces	31
Obrázek 11- Datová kostka	36
Obrázek 12 - EPC diagram hlavního procesu	43
Obrázek 13 - Struktura ACC informačního systému	44
Obrázek 14 - Proces provádění analýz	46
Obrázek 15 - Blokové schéma softwarové podpory	47
Obrázek 16 - Entito-relační diagram	49
Obrázek 17 - Model Hvězdy	50
Obrázek 18 - Připojení k SQL databázi	51
Obrázek 19 - Diagram načítání SQL instancí	52
Obrázek 20 - Diagram načtení databázi	53
Obrázek 21 - Kontrola vybrané databáze	54
Obrázek 22 - Ukládání záznamů	55
Obrázek 23 - Try blok	56
Obrázek 24 - Zobrazování záznamů	57
Obrázek 25 - Rozhraní pro filtraci	58
Obrázek 26 - Proces vyhledávání PSČ	60
Obrázek 27 - Přiřazování nových ID	61
Obrázek 28 - Připojení k XLS souboru	62
Obrázek 29 - Import záznamů z XLS	63
Obrázek 30 - Konverze formátů času	64
Obrázek 31 - Nahrávání záznamů do datového skladu	65
Obrázek 32 - Výsledky pro Jihočeský kraj	68
Obrázek 33 - Výsledky Středočeského kraje a kraje Vysočina	68

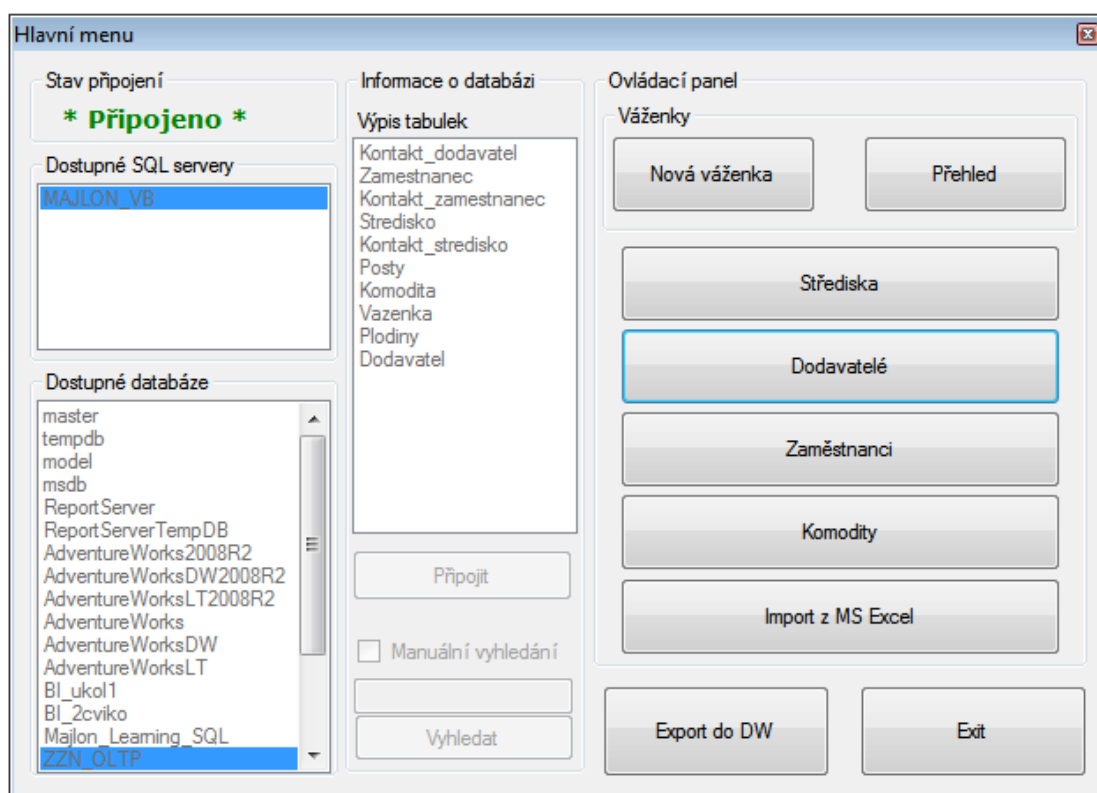
Seznam tabulek

Tabulka 1 - Porovnání OLTP a OLAP	23
Tabulka 2 - Porovnání relačních db a datových skladů	25
Tabulka 3: Seznam členů skupiny AGROFERT a.s.	41
Tabulka 4 - Struktura databáze partnerů	59
Tabulka 5 - Výstup MS BIDS studia	67
Tabulka 6 - Nejvhodnější partneři na nabídku dopravy	70

7. SEZNAM PŘÍLOH

(1) Ukázka práce s programem

Hlavní menu s výběrem správné databáze, hlavním ovládacím panelem a možností manuálního vyhledávání instancí SQL Serveru



Modul pro práci s entitou zaměstnanec (Vytváření nových záznamů, editace, prohlížení)

Zaměstnanci

Údaje

ID	Příjmení	Funkce
1	Příjmení 1	Silomistr
2	Příjmení 2	Silomistr
3	Příjmení 3	Silomistr
4	Příjmení 4	Silomistr
5	Příjmení 5	Silomistr
6	Příjmení 6	Silomistr
7	Příjmení 7	Silomistr
8	Příjmení 8	Silomistr
9	Příjmení 9	Silomistr
10	Příjmení 10	Silomistr
11	Příjmení 11	Silomistr
12	Příjmení 12	Silomistr
13	Příjmení 13	Silomistr
14	Příjmení 14	Silomistr

Filtrace
ID Příjmení PSC

Jméno ID
Jméno 1 1
Příjmení
Příjmení 1
Ulice ČP
Ulice A 1
Funkce Plat
Silomistr 10000
Město / Obec PSC
Město 1 39301 ✓
Kraj Okres
Vysočina Pelhřimov
Pošta Stát
Pelhřimov Česká Republika

Kontakty
Typ Kontakt

Nová registrace Editovat Přidat kontakt Zrušit

Menu pro vytvoření nového záznamu váženky. Uložení probíhá až po zadání všech potřebných informací.

Nová váženka

Dodavatel

IC	Název	PSC
0000		

IC dodavatele Název dodavatele PSC
0000

Komodita

ID	Název	Plodina
1	PSENICE POTRAVINARS...	Pšenice
2	PSENICE POTRAVINARS...	Pšenice
3	PSENICE POTRAVINARS...	Pšenice

Název komodity Plodina Odrůda

Středisko

ID	Název	Město
1	NS Pelhřimov Výrobní K...	Město 1
2	NS Pelhřimov	Město 2
3	NS Humpolec	Město 3

ID střediska Název Město

Zaměstnanec

ID	Příjmení	PSC
1	Příjmení 1	39301
2	Příjmení 2	39301
3	Příjmení 3	39601

ID zaměstnance Příjmení PSC

Ostatní informace
Datum Vlhkost [%] Brutto [Kg] Tara [Kg] Netto [Kg] SPZ
19. 5.2012

Uložit ☐ Napadení škůdci Dodavatel Komodita Středisko ☒ Zaměstnanec Stomo

Menu pro importování dat pro různé entity z formátu XLS.

The 'Import' dialog box features a tabbed interface with tabs for 'Dodavatelé', 'Pošty', 'Střediska', 'Váženky', 'Zaměstnanci', and 'Komodity'. The 'Váženky' tab is currently selected. On the left, there is a 'Načíst soubor' button, a checkbox for 'Hlásit chyby', and a section for 'Dostupné listy' which is currently empty. Below this is an 'Informace' section showing 'Počet sloupců:' and 'Počet řádků:'. The main area on the right is titled 'Mapování' and contains dropdown menus for 'IC', 'Název', 'Ulice', 'CP', 'PSC', and 'Město'. To the right of these are 'Import' and 'Zavřít' buttons. Below the mapping section is a large 'Data' area, which is currently empty.

Modul pro export dat do datového skladu. Počet ovládacích a informačních prvků byl minimalizován a bylo použito informační okno.

The 'export' dialog box displays 'Informace o datech v OLTP aplikaci' in a table format:

Počet záznamů dodavatelů: 1117	Počet záznamů váženek: 62539
Počet záznamů komodit: 116	Poslední ID váženky: 65475
Počet záznamů středisek: 17	Počet záznamů datového skladu:
Počet záznamů zaměstnanců: 17	

Below the table is a horizontal separator bar. Underneath is an 'Informační okno' containing two status messages: '*INF* Připojuji se k OLTP datbázi' and '*INF* Informace staženy OK'. At the bottom, there are two buttons: 'Spustit export' on the left and 'Zavřít' on the right.